



Controladores programáveis Micro830, Micro850 e Micro870

Códigos de catálogo do controlador Micro810 2080-LC10-12AWA, 2080-LC10-12QWB, 2080-LC10-12DWD, 2080-LC10-12QBB

Códigos de catálogo do controlador Micro820 2080-LC20-20AWB, 2080-LC20-20AWBR, 2080-LC20-20QWB, 2080-LC20-20QWBR, 2080-LC20-20QBB, 2080-LC20-20QBBR

Códigos de catálogo do controlador Micro830 2080-LC30-10QWB, 2080-LC30-10QVB, 2080-LC30-16AWB, 2080-LC30-16QWB, 2080-LC30-16QVB, 2080-LC30-24QWB, 2080-LC30-24QVB, 2080-LC30-24QBB, 2080-LC30-48AWB, 2080-LC30-48QWB, 2080-LC30-48QVB, 2080-LC30-48QBB

Códigos de catálogo do controlador Micro850 2080-LC50-24AWB, 2080-L50E-24AWB, 2080-LC50-24QWB, 2080-L50E-24QWB, 2080-LC50-24QVB, 2080-L50E-24QVB, 2080-LC50-24QBB, 2080-L50E-24QBB, 2080-LC50-48AWB, 2080-L50E-48AWB, 2080-LC50-48QWB, 2080-L50E-48QWB, 2080-LC50-48QWBK, 2080-L50E-48QWBK, 2080-LC50-48QVB, 2080-L50E-48QVB, 2080-LC50-48QBB, 2080-L50E-48QBB

Códigos de catálogo do controlador Micro870 2080-LC70-24AWB, 2080-L70E-24AWB, 2080-LC70-24QWB, 2080-L70E-24QWB, 2080-LC70-24QWBK, 2080-L70E-24QWBK, 2080-L70E-24QWBN, 2080-LC70-24QBB, 2080-L70E-24QBB, 2080-LC70-24QBBK, 2080-L70E-24QBBK, 2080-L70E-24QBBN



Allen-Bradley

by **ROCKWELL AUTOMATION**

Informações importantes para o usuário

Leia este documento e os documentos listados na seção de recursos adicionais sobre a instalação, configuração e operação deste equipamento antes de instalar, configurar, operar ou realizar a manutenção deste produto. Os usuários são obrigados a se familiarizar com as instruções de instalação e fiação, além das exigências de todos os códigos, leis e normas aplicáveis.

Atividades incluindo instalação, ajustes, colocação em serviço, utilização, montagem, desmontagem e manutenção devem ser efetuadas por pessoas com formação adequada, de acordo com o código de prática aplicável.

Se este equipamento for usado de forma não especificada pelo fabricante, a proteção fornecida pelo equipamento poderá ser prejudicada.

Em nenhuma hipótese, a Rockwell Automation será responsável por danos indiretos ou resultantes do uso ou da aplicação deste equipamento.

Os exemplos e diagramas apresentados neste manual são apenas para fins ilustrativos. Devido às diversas especificações e variáveis associadas a cada instalação específica, a Rockwell Automation, Inc. não pode assumir a responsabilidade pelo uso com base nos exemplos e diagramas.

A Rockwell Automation, Inc. não assume responsabilidade de patente quanto ao uso de informações, circuitos, equipamentos ou softwares descritos neste manual.

É proibida a reprodução, parcial ou total, deste manual sem a permissão por escrito da Rockwell Automation, Inc.

Ao longo deste manual, quando necessário, usamos observações para alertá-lo sobre as considerações de segurança.



ADVERTÊNCIA: Identifica as informações sobre práticas ou circunstâncias que possam causar explosão numa área classificada, resultando em ferimentos pessoais ou morte, prejuízos a propriedades ou perdas econômicas.



ATENÇÃO: Identifica as informações sobre práticas ou circunstâncias que podem causar ferimentos pessoais ou morte, prejuízos à propriedade ou perdas econômicas. O símbolo de atenção ajuda você a identificar e evitar um perigo e reconhecer as consequências.

IMPORTANTE Identifica informações importantes para a correta aplicação e compreensão do produto.

As etiquetas também podem estar sobre ou dentro do equipamento para fornecer precauções específicas.



PERIGO DE CHOQUE: As etiquetas podem estar sobre ou dentro do equipamento (por exemplo, um inversor ou motor) para alertar as pessoas da presença de tensão perigosa.



PERIGO DE QUEIMADURA: As etiquetas podem estar sobre ou dentro do equipamento (por exemplo, um inversor ou motor) para alertar as pessoas que as superfícies podem atingir temperaturas perigosas.



RISCO DE ARCO ELÉTRICO: As etiquetas podem estar no equipamento ou dentro dele, por exemplo, em um centro de controle de motores, para alertar as pessoas quanto a arco elétrico potencial. Arcos elétricos causam ferimentos graves ou morte. Utilize equipamentos de proteção individual (EPI) adequados. Siga TODOS os requisitos regulatórios quanto a práticas de trabalho seguro e em relação aos equipamentos de proteção individual (EPI).

Prefácio

Sobre esta publicação	11
Catálogos com proteção contra ambientes agressivos	11
Faça o download do firmware, Add-on Profile, EDS e outros arquivos	11
Resumo das alterações	11
Recursos adicionais	12

Características gerais de hardware

Capítulo 1

Recursos do hardware	16
Micro830 Controladores	16
Controladores Micro850	17
Controladores Micro870	19
Cabos de programação	21
Cabos para porta serial incorporada	22
Suporte Ethernet incorporado	22

Sobre o controlador

Capítulo 2

Software de programação para controladores Micro800	23
Obtenha o Software Connected Components Workbench	23
Use o Software Connected Components Workbench	23
Alterações do controlador no modo de operação	23
Usar alteração do modo de operação (RMC)	24
Alterações livres	25
Memória RMC	26
Limitações do RMC	27
Usando a alteração do modo de operação (RMCC)	28
Uso da comunicação Modbus RTU	29
Usando comunicação EtherNet/IP	31
Considerações de segurança	32
Desconectar a alimentação principal	33
Circuitos de segurança	33
Distribuição de potência	33
Testes periódicos do Circuito do relé de controle mestre	33
Considerações sobre alimentação	34
Transformadores de isolamento	34
Energização da fonte de alimentação	34
Perda da fonte de alimentação	34
Estados de entrada ao cortar a alimentação	35
Outros tipos de condições da linha	35
Prevenindo aquecimento excessivo	35
Relé de controle mestre	35
Utilizando chaves de parada de emergência	36

Capítulo 3

Instale seu controlador

Dimensões de montagem do controlador	39
Dimensões de montagem	39

Montagem em trilho DIN	41
Montagem em painel	41
Dimensões da montagem em painel	42
Montagem do sistema	43

Capítulo 4

Fiação do controlador

Especificação da fiação e recomendações	47
Uso de supressores de transiente	48
Supressores de transientes recomendados	49
Aterrando o controlador	50
Esquemas elétricos	51
Fiação de E/S do controlador	55
Minimização do ruído elétrico	55
Orientações para fiação de canal analógico	55
Minimização do ruído elétrico nos canais analógicos	56
Aterrando o cabo analógico	56
Exemplos de fiação	57
Fiação para porta serial incorporada	58

Capítulo 5

Conexões de comunicação

Visão geral	59
Protocolos de comunicação suportados	59
Modbus RTU	61
Cliente/servidor CIP serial – DF1	61
ASCII	61
Cliente/Servidor Modbus/TCP	61
Cliente/servidor CIP Symbolic	62
Mensagem de cliente CIP	63
Soquetes cliente/servidor TCP/UDP	64
Transmissão de comunicação CIP	64
Exemplos de arquiteturas suportadas	64
Usar modems com controladores Micro800	65
Realizando uma conexão DF1 ponto a ponto	65
Fabrique o seu próprio cabo para modem	66
Configuração da porta serial	66
Configurar o driver serial CIP	66
Configuração Modbus RTU	70
Configurar ASCII	71
Configurar as definições de Ethernet	72
Validar endereço IP	74
Nome do host Ethernet	74
Configurar o driver serial CIP	74
Suporte OPC usando FactoryTalk Linx	75

Capítulo 6

Protocolo de rede distribuída do controlador Micro870

Visão geral	77
Configuração de canal para escravo DNP3	77
Configuração da camada do link da porta serial	78
Configuração da camada Ethernet	79
Configuração da camada de aplicação escrava DNP3	80

Parâmetros de configuração da camada do link serial	80
Parâmetros de configuração da camada Ethernet	83
Parâmetros de configuração da camada de aplicação escrava DNP3.	88
Camada de aplicação escrava DNP3	103
Códigos de função	103
Indicações internas	107
Objetos DNP3 e variáveis do controlador	108
Dados do objeto DNP3.	109
Configuração do DNP3	110
Objeto de conjunto de dados DNP3.	110
Flags de qualidade do objeto	117
Objeto de atributo de dispositivo DNP3	119
Relatório de eventos	121
Gerar eventos	121
Registro de eventos DNP3 10K	123
Controle de geração de eventos.	124
Evento de relatório por resposta com coleta.	124
Evento de relatório por resposta não solicitada	125
Prevenção de colisão	126
Sincronização de Tempo	126
Diagnóstico	128
Diagnóstico para canal Ethernet.	129
Diagnóstico para autenticação segura	130
Códigos de função	130
Tabela de implementação	131

Capítulo 7

Execução de programa no Micro800

Visão geral da execução de programas	137
Regras para execução	138
Módulo opcional	139
Considerações sobre carga e desempenho do controlador	139
Execução periódica de programas	139
Energização e primeira varredura	140
Retenção de variável	140
Alocação de memória	141
Orientações e limitações para usuários avançados.	141

Controle de movimento**Capítulo 8**

Controle de movimento PTO.	143
Use o recurso de controle de movimento do Micro800	145
Sinais de entrada e saída.	146
Blocos de funções de controle de movimento.	149
Regras gerais para os Blocos de funções de controle de movimento	150
Movimento do eixo e parâmetros	158
Estados de eixo	159
Limites.	160
Parada de movimento	162
Direção do movimento	163
Elementos de eixo e tipos de dados.	164
Possibilidades de erro no eixo.	165
Tipo de dado MC_Engine_Diag	166
Códigos de erro de bloco de funções e status de eixo	166
Gerenciamento de falha grave	169
Configuração do eixo de movimento no Connected Components Workbench	169
Adicionar um novo eixo.	170
Editar a configuração de eixo	171
Velocidade inicial/parada do eixo	177
Resolução de dados reais.	177
Precisão de pulso PTO	179
Validação de parâmetros de eixo de movimento	180
Excluir um eixo	180
Monitorar um eixo	180
Bloco de funções de retorno à posição inicial	181
Condições para retorno bem-sucedido à posição inicial	182
MC_HOME_ABS_SWITCH	182
MC_HOME_LIMIT_SWITCH	184
MC_HOME_REF_WITH_ABS	185
MC_HOME_REF_PULSE	186
MC_HOME_DIRECT	188
Use a PTO para controle PWM	188
POU PWM_Program	189
Eixo de realimentação HSC	190

Capítulo 9**Use o contador de alta velocidade e a chave de fim de curso programável**

Características gerais do contador de alta velocidade	191
Características gerais da chave de fim de curso programável	191
O que é um contador de alta velocidade?	192
Recursos e operação	192
Entradas HSC e mapeamento da fiação	193
Estruturas de dados do contador de alta velocidade (HSC)	196
Estrutura de dados HSC APP	196
Estrutura de dados STS HSC (Status HSC).	205
Bloco de funções HSC (Contador de alta velocidade)	211
Comandos HSC (HScCmd)	212
Bloco de funções HSC_SET_STS.	213
Função de chave de fim de curso programável (PLS)	214

Estrutura de dados PLS	214
Operação PLS	215
Exemplo PLS	216
Interrupções HSC	217
Configuração de interrupção HSC	217
POU da interrupção HSC	218
Informação de status de interrupção HSC	219

Capítulo 10

Segurança do controlador

Modo protegido	221
Acesso exclusivo	221
Proteção por senha	222
Compatibilidade	222
Trabalhar com um controlador bloqueado	223
Carregar a partir de um controlador protegido com senha	223
Depurar um controlador protegido por senha	224
Download para um controlador protegido por senha	224
Transfira o programa do controlador e proteja com senha o controlador que receberá os dados	224
Fazer backup e restaurar um controlador protegido por senha ..	225
Configurar a senha do controlador	226
Recuperação de uma senha perdida	226
O projeto no controlador será perdido mas pode-se fazer download de um novo projeto.	226

Capítulo 11

Usando cartões microSD

Características gerais	229
Fazer backup e restaurar projeto	230
Backup e restauração da estrutura do diretório	231
Ajustes de parâmetros de energização no ConfigMeFirst.txt. ...	232
Regras gerais de configuração no ConfigMeFirst.txt	234
Erros do ConfigMeFirst.txt.	234
Entregar atualizações de projetos aos clientes por e-mail	235
Registro de dados	237
Estrutura do diretório de registro de dados	238
Bloco de funções de registro de dados (DLG)	239
Receita	243
Estrutura do diretório de receita	244
Projetos de início rápido para blocos de funções de registro de dados e de receita	247
Usar o recurso de registro de dados	247
Use o recurso Receita	252

Apêndice A

Mapeamento Modbus para Micro800

Mapeamento Modbus	259
Configuração Endian	259
Espaço de endereço de mapeamento e tipos de dados suportados.	259
Exemplo 1, PanelView 800 IHM (mestre) para Micro800 (escravo)	260

Exemplo 2, Micro800 (mestre) para inversor de frequência PowerFlex 4M (escravo)	261
Desempenho	264

Apêndice B

Início rápido

Faça o upgrade flash do firmware de seu Micro800	265
Atualização flash a partir do cartão MicroSD	267
Estabeleça comunicações entre RSLinx e um controlador Micro830/Micro850/Micro870 por USB	271
Configure a senha do controlador.	275
Defina a senha do controlador	276
Mude a senha.	277
Remova a senha.	278
Use o contador em alta velocidade	278
Criar o projeto HSC e as variáveis.	279
Atribuir valores às variáveis HSC	282
Atribuir variáveis ao bloco de funções	285
Execute o contador em alta velocidade	286
Use a função de chave de fim de curso programável (PLS)	288
Forçando E/S	289
Verificar se as forças (travas) estão habilitadas	289
Force de E/S após um ciclo de energização/desenergização	290
Usar alteração do modo de operação	291
Criação do projeto	291
Edite o projeto usando Alterar o modo de operação	293

Apêndice C

Interrupções do usuário

Informações sobre o uso de interrupções	297
O que é uma interrupção?	297
Quando se pode interromper uma operação do controlador?	298
Prioridade de interrupções do usuário	298
Configuração de interrupções do usuário.	299
Rotina de falha do usuário	300
Instruções sobre interrupções do usuário	300
STIS – Início selecionado em função do tempo	301
UID – Desabilitar interrupção do usuário	301
UIE – Habilitar interrupção do usuário	303
UIF – Fluxo de interrupção do usuário	304
UIC – Remoção de interrupção do usuário.	305
Usando a função de interrupção selecionada em função do tempo (STI)	306
Configuração e status da interrupção selecionada em função do tempo (STI).	307
Configuração da função STI	307
Informações de status da função STI	307
Usando a função de interrupção de entrada de evento (EII).	309
Configuração e status da função de interrupção de entrada de evento (EII)	309
Configuração da função EII	309
Informações de status da função EII	310

Localização de falhas**Apêndice D**

Indicadores de status no controlador	313
Operação normal	315
Códigos de erro	315
Tipos de falha	315
Ação corretiva para falhas recuperáveis e não recuperáveis	321
Recuperar um registro de falha	321
Modelo de recuperação de erro do controlador	321
Ligar para a Rockwell Automation para obter assistência	322

Blocos de função PID**Apêndice E**

Bloco de funções PID	324
Bloco de funções IPIDCONTROLLER	326
Como fazer o ajuste automático	328
Como o Autotune funciona	329
Localização de falhas no processo autotune	329
Exemplo de aplicação PID	330
Exemplo de código PID	331

Carregamento do sistema**Apêndice F**

Calcule a energia total para seu controlador Micro830/Micro850/ Micro870	333
---	-----

Conexão a redes usando DF1**Apêndice G**

Protocolo DF1 full-duplex	335
Protocolo DF1 half-duplex	336
Operação DF1 half-duplex	336
Considerações ao se comunicar como um DF1 escravo em um link multiponto	337
Uso de modems com controladores programáveis Micro800 ...	337
Operação da linha de controle do modem	338
DF1 full-duplex	338
DF1 half-duplex escravo	338
DF1 half-duplex mestre	339
Modem de rádio DF1	339
Configurar parâmetros DF1 half-duplex	340
RTS Send Delay e RTS Off Delay	340
Configurar uma estação DF1 half-duplex mestre de modo padrão ..	341
Tempo limite mínimo de ACK do DF1 half-duplex mestre	343
Determinação do tempo limite mínimo de ACK do mestre	344
Diagnóstico de comunicação do DF1 half-duplex mestre	345
Configure um modo DF1 half-duplex baseado em mensagem Estação mestre	345
Configurar uma estação escrava	348
Configurar o tempo limite de coleta	349
Diagnóstico de comunicação do DF1 half-duplex escravo	350

Configurar uma estação de modem de rádio	350
Diagnóstico de comunicação do modem de rádio DF1	351
Configurar a tabela de armazenamento e encaminhamento	352
Índice	355

Sobre esta publicação

Use este manual se você for responsável pelo projeto, instalação, programação ou sistemas de controle de localização de falhas que usam os controladores Micro800™.

Você deve ter uma compreensão básica de circuitos elétricos e familiaridade com lógica ladder. Se não o tiver, obtenha o treinamento adequado antes de usar este produto.

Este manual é um guia de referência para módulos plug-in e acessórios dos controladores Micro800. Ele descreve os procedimentos que são usados para instalar, conectar e localizar as falhas de seu controlador. Este manual:

- explica como instalar e conectar seus controladores
- fornece as características gerais do sistema do controlador Micro800

Consulte a ajuda on-line fornecida com o software Connected Components Workbench™ para mais informações sobre a programação do seu controlador Micro800.

A Rockwell Automation reconhece que alguns dos termos que são usados atualmente em nosso setor e nesta publicação não estão alinhados com o movimento em direção a uma linguagem inclusiva na área de tecnologia. Estamos colaborando proativamente com os colegas do setor para encontrar alternativas a tais termos e fazer alterações em nossos produtos e conteúdo. Pedimos desculpas pelo uso de tais termos em nosso conteúdo enquanto implementamos essas alterações.

Catálogos com proteção contra ambientes agressivos

Os códigos de catálogo com o sufixo “K” são os que têm proteção contra ambientes agressivos e suas especificações são as mesmas que os demais catálogos.

Faça o download do firmware, Add-on Profile, EDS e outros arquivos

Faça o download do firmware, arquivos associados (tais como Add-on Profile, EDS e DTM) e acesse as notas da versão do produto no Centro de Download e Compatibilidade do Produto em rok.auto/pcdc.

Resumo das alterações

Esta publicação contém as seguintes informações novas ou atualizadas. Esta lista inclui somente atualizações relevantes e não se destina a refletir todas as alterações.

Tópico	Página
Adicionados novos catálogos, Micro850 (2080-L50E) e Micro870 (2080-L70E)	Em todo o manual
Atualizados os Recursos adicionais	12
Atualizada a tabela de Controladores Micro850 – número e tipos de entradas/saídas	20
Atualizada a tabela de Controladores Micro870 – número e tipos de entradas/saídas	21

Tópico	Página
Atualizada a seção Use o software Connected Components Workbench	23
Atualizado o tópico Usando a alteração do modo de operação (RMC)	24
Atualizada a seção Memória RMC	26
Atualizada a seção Limitações do RMC	27
Atualizado o tópico Usando a alteração da configuração do modo de operação (RMCC)	28
Atualizada a seção CIP Serial Cliente/Servidor – DF1	61
Atualizada a seção Cliente/Servidor CIP Symbolic	62
Atualizada a Tabela de parâmetros do driver Serial CIP	67
Novo capítulo Protocolo de rede distribuída do controlador Micro870	77
Atualizado o tópico Proteção por senha	222
Atualizada a seção Faça o backup e restaure um controlador protegido por senha	225
Atualizado o tópico Usando o plug-in do módulo de memória	226
Atualizado o tópico Usando os cartões microSD, Visão geral	229
Atualizado o tópico Backup e restauração de projeto	230
Removidas Especificações do apêndice. As especificações para os controladores Micro800 podem ser encontradas em Micro800 Programmable Controllers, Technical Data, publicação 2080-TD001	-
Adicionado o apêndice Conecte a redes usando DF1	335

Recursos adicionais

Estes documentos contêm informações adicionais em relação a produtos relacionados da Rockwell Automation.

Recurso	Descrição
Família de Controladores Programáveis, Micro800, Guia de seleção, publicação 2080-SG001	Fornece informações para ajudar você a selecionar o controlador Micro800, os plug-ins, a expansão de E/S e os acessórios certos, de acordo com seus requisitos.
Micro800 Programmable Controllers, Technical Data, publicação 2080-TD001	Fornece especificações detalhadas para controladores Micro800, módulos de expansão de E/S, módulos plug-in e acessórios.
Módulos de expansão de E/S do Micro800, Manual do usuário, publicação 2080-UM003	Informações sobre recursos, configuração, fiação, instalação e especificações para os módulos Expansion I/O Micro800 e fonte de alimentação.
Micro800 Plug-in Modules, User Manual, publicação 2080-UM004	Informações sobre recursos, configuração, instalação, fiação e especificações para os módulos plug-in Micro800.
Instruções gerais para controladores programáveis Micro800, Manual de referência, publicação 2080-RM001	Informações sobre conjuntos de instruções para desenvolver programas em sistemas de controle Micro800.
Controladores programáveis Micro800: Introdução ao controle de movimento usando um eixo simulado, Guia rápido, publicação 2080-QS001	Fornece instruções para iniciar rapidamente a implementação do projeto de controle de movimento no software Connected Components Workbench.
Controladores programáveis Micro800: Iniciando com a mensagem de cliente CIP, Início Rápido, publicação 2080-QS002	Fornece instruções para iniciar rapidamente a usar o envio de mensagens CIP GENERIC e CIP Symbolic.
Controladores programáveis Micro800: Introdução ao PanelView Plus, Guia Rápido, publicação 2080-QS003	Fornece instruções para iniciar rapidamente a usar variáveis globais para controladores Micro800 juntamente com os terminais PanelView™ Plus IHM.
Configuring Micro800 Controllers on FactoryTalk Gateway, Quick Start, publicação 2080-QS005	Fornece instruções para iniciar rapidamente a configurar um controlador Micro800 no FactoryTalk Linx Gateway.
Kinetix 3 Motion Control Indexing Device Building Block – Connected Components Accelerator Toolkit, Quick Start, publicação CC-QS025	Fornece instruções para início rápido para implementar uma aplicação de indexação de inversor de frequência Kinetix 3 usando o Software Connected Components Workbench e um controlador Micro800.
Motion Control PTO Application Building Block – Connected Components Accelerator Toolkit, Quick Start, publicação CC-QS033	Fornece instruções para início rápido para implementar controle de movimento PTO de um inversor de frequência Kinetix 3 usando o Software Connected Components Workbench e um controlador Micro800.
Fonte de alimentação CA externa do controlador programável Micro800™, Instruções de instalação 2080-IN001	Informações sobre a montagem e a fiação da fonte de alimentação opcional externa.
Micro800 Programmable Controllers, Installation Instructions, publicação 2080-IN013	Informações sobre a montagem e fiação dos controladores Micro800.
Módulos de entrada com entrada/saída de 12/24 V de 16 e 32 pontos Micro800, Instruções de instalação, publicação 2085-IN001	Informações sobre a montagem e a fiação dos módulos de expansão de E/S (2085-IQ16, 2085-IQ32T).
Módulo terminador de via Micro800™, Instruções de instalação, publicação 2085-IN002	Informações sobre a montagem e a fiação do terminador de via de expansão de E/S (2085-ECR).
Módulos de saída de 12/24 Vcc com saída de 16 pontos e entrada de corrente de 16 pontos Micro800, Instruções de instalação, publicação 2085-IN003	Informações sobre a montagem e a fiação dos módulos de expansão de E/S (2085-OB16, 2085-OB16).

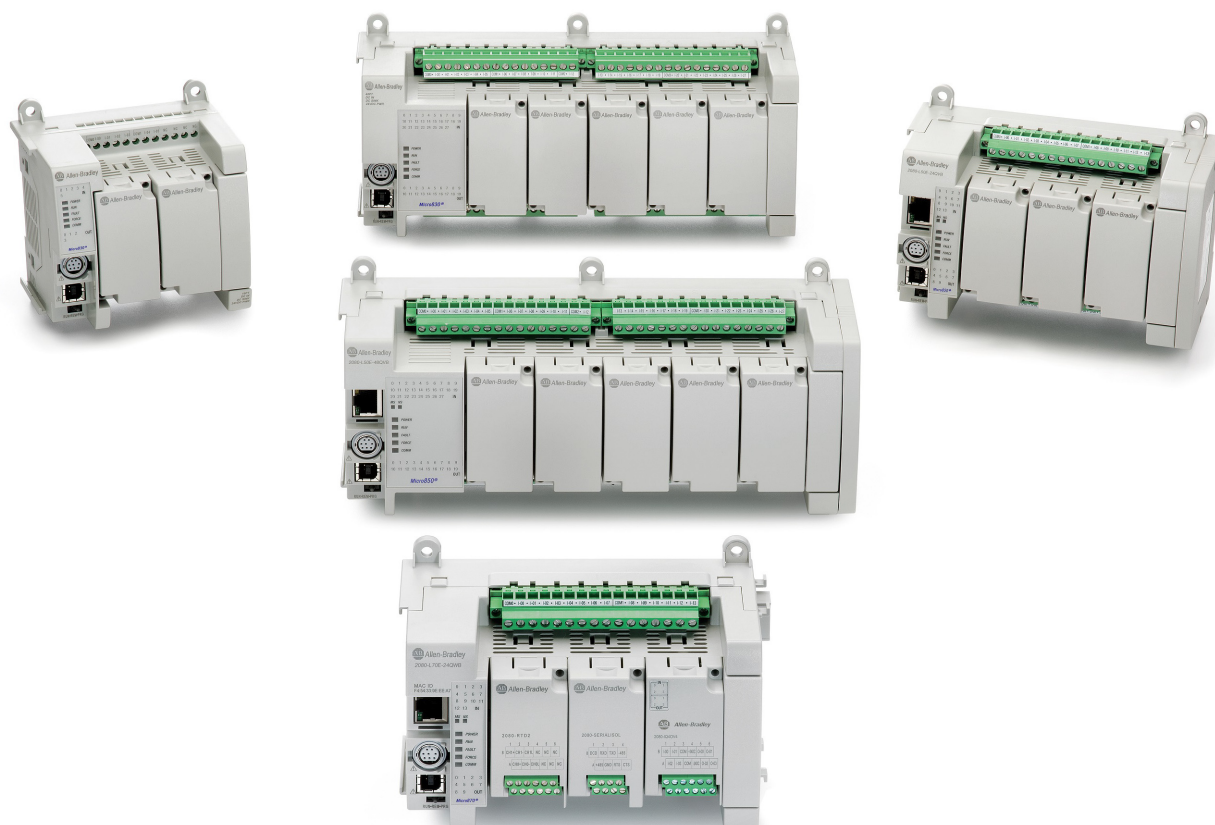
Recurso	Descrição
Módulos de saída a relé CA/CC de 8 e 16 pontos Micro800, Instruções de instalação, publicação 2085-IN004	Informações sobre a montagem e a fiação dos módulos de expansão de E/S (2085-0W8, 2085-0W16).
Módulos de CA de entrada com entrada de 8 pontos e saída de 8 pontos Micro800, Instruções de instalação, publicação 2085-IN005	Informações sobre a montagem e a fiação dos módulos de expansão de E/S (2085-IA8, 2085-IM8, 2085-0A8).
Módulos de entrada e saída de tensão/corrente analógica de 4 e 8 canais Micro800, Instruções de instalação, publicação 2085-IN006	Informações sobre a montagem e a fiação dos módulos de expansão de E/S (2085-IF4, 2085-IF8, 2085-0F4).
Termopar de 4 canais/módulo de entrada RTD Micro800, Instruções de instalação, publicação 2085-IN007	Informações sobre a montagem e a fiação do módulo de expansão de E/S (2085-IRT4).
Controladores programáveis Micro870 com fonte de alimentação de expansão de 24 Vcc, Instruções de instalação, publicação 2085-IN008	Informações sobre a montagem e a fiação da fonte de alimentação opcional externa para módulos de expansão de E/S.
Módulo de encaixe da porta serial isolada RS-232/RS-485 Micro800™, Esquemas elétricos, publicação 2080-WD002	Informações sobre a montagem e a fiação do módulo de encaixe da porta serial isolada RS-232/RS-485 do Micro800.
Módulo de encaixe de entrada analógica unipolar não isolada Micro800™, Esquemas elétricos, publicação 2080-WD003	Informações sobre a montagem e a fiação do módulo plug-in de entrada analógica unipolar não isolada Micro800.
Módulo de encaixe de saída analógica unipolar não isolada Micro800™, Esquemas elétricos, publicação 2080-WD004	Informações sobre a montagem e a fiação do módulo plug-in de saída analógica unipolar não isolada Micro800.
Módulo de encaixe RTD não isolado Micro800™, Esquemas elétricos, publicação 2080-WD005	Informações sobre a montagem e a fiação do módulo plug-in RTD não isolado Micro800.
Módulo de encaixe do termopar não isolado Micro800™, Esquemas elétricos, publicação 2080-WD006	Informações sobre a montagem e a fiação do módulo plug-in termopar não isolado Micro800.
Módulo plug-in RTC de alta precisão e backup de memória Micro800, Esquemas elétricos, publicação 2080-WD007	Informações sobre a montagem e a fiação do módulo plug-in RTC de alta precisão e backup de memória Micro800.
Módulo de encaixe de entrada analógica do potenciômetro trim de 6 canais Micro800™, Esquemas elétricos, publicação 2080-WD008	Informações sobre a montagem e a fiação do módulo plug-in de entrada analógica de ajuste de 6 canais Micro800.
Micro800 Digital Relay Output Plug-in Module, Wiring Diagrams, publicação 2080-WD010	Informações sobre a montagem e a fiação do módulo plug-in de saída a relé digital do Micro800.
Micro800 Digital Input, Output, and Combination Plug-in Modules, Wiring Diagrams, publicação 2080-WD011	Informações sobre a montagem e a fiação dos módulos plug-in de combinação, saída e entrada digital Micro800.
Módulo de encaixe com contador em alta velocidade Micro800, Esquemas elétricos, publicação 2080-WD012	Informações sobre como montar e ligar por cabo o módulo plug-in do contador de alta velocidade.
Módulo de encaixe Micro800 DeviceNet, Esquemas elétricos, publicação 2080-WD013	Informações sobre como montar e ligar por cabo o módulo de encaixe DeviceNet® do Micro800.
EtherNet/IP Network Devices, User Manual, publicação ENET-UM006	Descreve como configurar e usar os dispositivos EtherNet/IP para se comunicar com a rede EtherNet/IP.
Ethernet Reference Manual, publicação ENET-RM002	Descreve conceitos básicos de Ethernet, componentes de infraestrutura e recursos de infraestrutura.
System Security Design Guidelines Reference Manual, publicação SECURE-RM001	Fornece orientação sobre como conduzir avaliações de segurança, implementar produtos da Rockwell Automation em um sistema seguro, consolidar o sistema de controle, gerenciar o acesso do usuário e descartar os equipamentos.
Manutenção preventiva de componentes industriais, gabinetes e especificações de classificações de contato, publicação IC-TD002	Fornece uma ferramenta de referência rápida para conjuntos e controles de automação industrial Allen-Bradley®.
Orientações de segurança para a aplicação, instalação e manutenção de controle de estado sólido, publicação SGI-1.1	Criado para harmonização com a publicação dos Padrões NEMA nº ICS 1.1-1987 e fornece orientações gerais para a aplicação, instalação e manutenção de controle de estado sólido na forma de dispositivos individuais ou conjuntos de pacotes incorporando componentes de estado sólido.
Orientação sobre fiação de automação industrial e aterramento, publicação 1770-4.1	Fornece orientações gerais para instalar um sistema industrial Rockwell Automation®.
Site de certificações de produto, rok.auto/certifications	Fornece declarações de conformidade, certificados e outros detalhes de certificação.

Você pode visualizar ou fazer download das publicações em [rok.auto/literature](#).

Você pode fazer download da versão mais recente do software Connected Components Workbench para seu controlador Micro800 em [rok.auto.ccw](#).

Observações:

Características gerais de hardware



Este capítulo fornece as características gerais dos recursos de hardware dos controladores Micro830®, Micro850® e Micro870®. Ele possui os seguintes tópicos:

Tópico	Página
Recursos do hardware	16
Micro830 Controladores	16
Controladores Micro850	17
Controladores Micro870	19
Cabos de programação	21
Cabos para porta serial incorporada	22
Suporte Ethernet incorporado	22

Recursos do hardware

Micro830, os controladores Micro850 e Micro870 são controladores robustos e econômicos, com entradas e saídas incorporadas. Dependendo do tipo de controlador, ele pode acomodar de dois a cinco módulos de encaixe. Os controladores Micro850 e Micro870 possuem recursos expansíveis. O controlador Micro850 pode suportar até quatro módulos de expansão de E/S e o controlador Micro870 pode suportar até oito módulos de expansão de E/S.

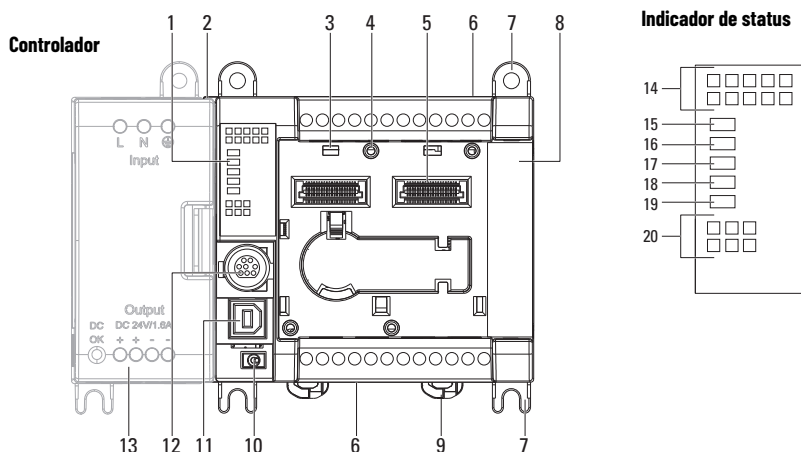
- IMPORTANTE** Para informações sobre módulos de encaixe suportados e Expansion I/O, consulte as seguintes publicações:
- Módulos de expansão de E/S do Micro800, Manual do usuário, publicação [2080-UM003](#)
 - Micro800 Plug-in Modules, User Manual, publicação [2080-UM004](#)

Os controladores também acomodam qualquer fonte de alimentação de saída de 24 Vcc classificada como Classe 2 que atenda às especificações mínimas, assim como a fonte de alimentação opcional Micro800.

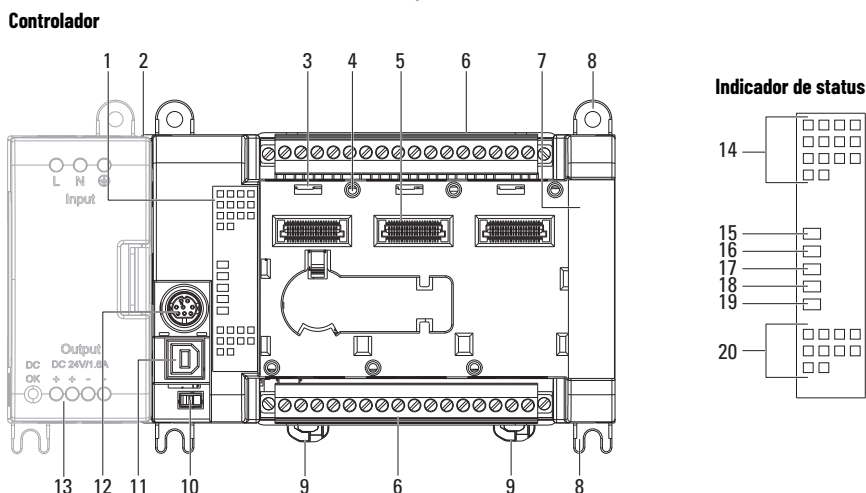
Consulte [Localização de falhas na página 313](#) para obter descrições sobre a operação do indicador de status para fins de localização de falhas.

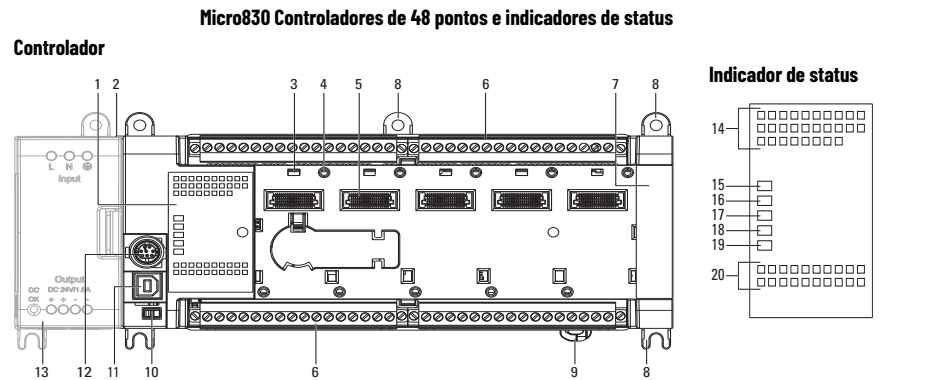
Micro830 Controladores

Micro830 Controladores de 10/16 pontos e indicadores de status



Micro830 Controladores de 24 pontos e indicadores de status





Descrição do controlador

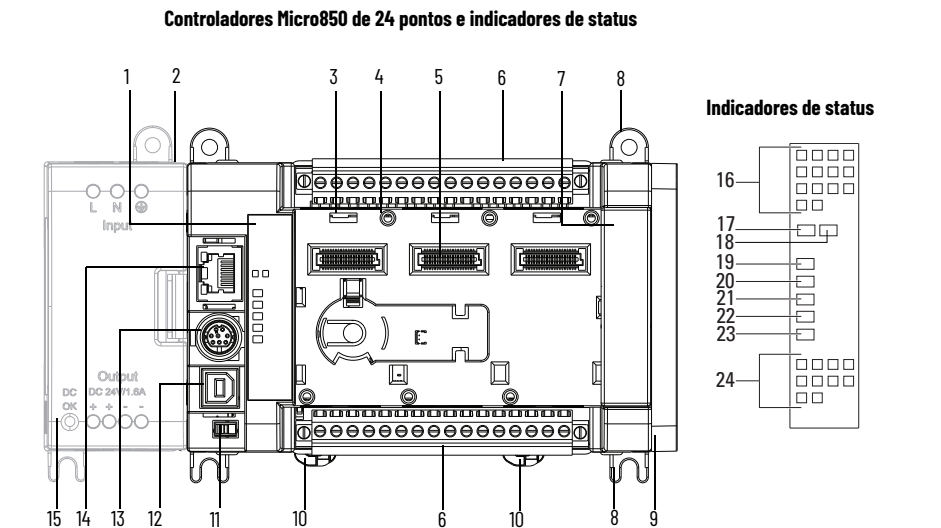
	Descrição		Descrição
1	Indicadores de status	8	Furo do parafuso de fixação/pé de montagem
2	Slot da fonte de alimentação opcional	9	Trava de montagem em trilho DIN
3	Trava plug-in	10	Chave de modo
4	Furo do parafuso de encaixe	11	Porta USB do conector tipo B
5	Conector de encaixe de alta velocidade com 40 pinos	12	Porta serial combinada não isolada RS-232/RS-485
6	Borne de E/S removível	13	Fonte de alimentação CA opcional
7	Tampa direita		

Descrição do Indicador de status⁽¹⁾

	Descrição		Descrição
14	Status de entrada	18	Status de force
15	Status de alimentação	19	Status de comunicação serial
16	Status de operação	20	Status de saída
17	Status de falha		

(1) Para uma descrição detalhada sobre os diferentes indicadores LED de status, consulte [Localização de falhas na página 313](#).

Controladores Micro850



Descrição do controlador

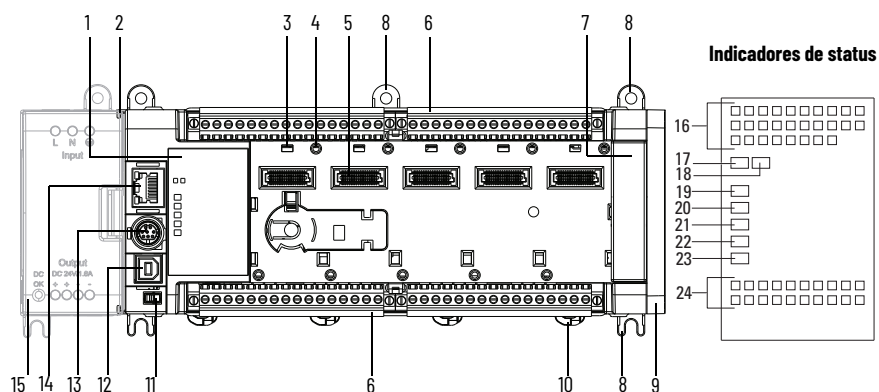
	Descrição		Descrição
1	Indicadores de status	9	Tampa do slot expansão de E/S
2	Slot da fonte de alimentação opcional	10	Trava de montagem em trilho DIN
3	Trava plug-in	11	Chave de modo
4	Furo do parafuso de encaixe	12	Porta USB do conector tipo B
5	Conector de encaixe de alta velocidade com 40 pinos	13	Porta serial combinada não isolada RS-232/RS-485
6	Borne de E/S removível	14	Conector Ethernet RJ-45 (com indicadores LED verdes e amarelos incorporados)
7	Tampa direita	15	Fonte de alimentação opcional
8	Furo do parafuso de fixação/pé de montagem		

Descrição do Indicador de status⁽¹⁾

	Descrição		Descrição
16	Status de entrada	21	Status de falha
17	Status do módulo	22	Status de force
18	Network Status	23	Status de comunicação serial
19	Status de alimentação	24	Status de saída
20	Status de operação		

(1) Para descrições detalhadas sobre os diferentes indicadores LED de status, consulte [Localização de falhas na página 313](#).

Controladores Micro850 de 48 pontos e indicadores de status



Descrição do controlador

	Descrição		Descrição
1	Indicadores de status	9	Tampa do slot expansão de E/S
2	Slot da fonte de alimentação opcional	10	Trava de montagem em trilho DIN
3	Trava plug-in	11	Chave de modo
4	Furo do parafuso de encaixe	12	Porta USB do conector tipo B
5	Conector de encaixe de alta velocidade com 40 pinos	13	Porta serial combinada não isolada RS-232/RS-485
6	Borne de E/S removível	14	Conector Ethernet/IP RJ-45 (com LEDs indicadores verdes e amarelos incorporados)
7	Tampa direita	15	Fonte de alimentação CA opcional
8	Furo do parafuso de fixação/pé de montagem		

Descrição do Indicador de status⁽¹⁾

	Descrição		Descrição
16	Status de entrada	21	Status de falha
17	Status do módulo	22	Status de força
18	Status da rede	23	Status de comunicação serial
19	Status de alimentação	24	Status de saída
20	Status de operação		

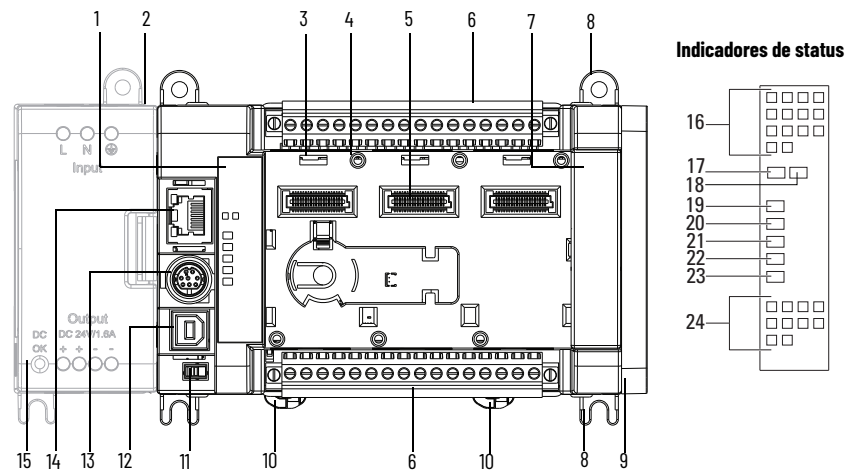
(1) Para descrições detalhadas sobre esses diferentes indicadores LED de status, consulte [Localização de falhas na página 313](#).

Observação: Você pode solicitar os bornes de substituição a seguir separadamente:

- 2080-RPL24RTB para controladores de base de 24 pontos
- 2080-RPL48RTB para controladores de base de 48 pontos

Controladores Micro870

Controladores Micro870 de 24 pontos e indicadores de status



Descrição do controlador

	Descrição		Descrição
1	Indicadores de status	9	Tampa do slot expansão de E/S
2	Slot da fonte de alimentação opcional	10	Trava de montagem em trilho DIN
3	Trava plug-in	11	Chave de modo
4	Furo do parafuso de encaixe	12	Porta USB do conector tipo B
5	Conector de encaixe de alta velocidade com 40 pinos	13	Porta serial combinada não isolada RS-232/RS-485
6	Borne de E/S removível	14	Conector Ethernet RJ-45 (com indicadores LED verdes e amarelos incorporados)
7	Tampa direita	15	Fonte de alimentação opcional
8	Furo do parafuso de fixação/pé de montagem		

Descrição do indicador de status⁽¹⁾

	Descrição		Descrição
16	Status de entrada	21	Status de falha
17	Status do módulo	22	Status de force
18	Network Status	23	Status de comunicação serial
19	Status de alimentação	24	Status de saída
20	Status de operação		

(1) Para descrições detalhadas sobre os diferentes indicadores LED de status, consulte [Localização de falhas na página 313](#).



Você pode solicitar bornes de substituição, código de catálogo 2080-RPL24RTB, separadamente.

Tabela 1 - Micro830 Controladores - número e tipos de entradas e saídas

Código de catálogo	Entradas		Saídas			Suporte a PTO	Suporte a HSC
	110 Vca	24 Vcc/Vca	Relé	Tipo sink de 24 V	Source de 24 V		
2080-LC30-10QWB	-	6	4	-	-	-	2
2080-LC30-10QVB	-	6	-	4	-	1	2
2080-LC30-16AWB	10	-	6	-	-	-	-
2080-LC30-16QWB	-	10	6	-	-	-	2
2080-LC30-16QVB	-	10	-	6	-	1	2
2080-LC30-24QWB	-	14	10	-	-	-	4
2080-LC30-24QVB	-	14	-	10	-	2	4
2080-LC30-24QBB	-	14	-	-	10	2	4
2080-LC30-48AWB	28	-	20	-	-	-	-
2080-LC30-48QWB	-	28	20	-	-	-	6
2080-LC30-48QVB	-	28	-	20	-	3	6
2080-LC30-48QBB	-	28	-	-	20	3	6

Tabela 2 - Micro850 Controladores - número e tipos de entradas e saídas

Código de catálogo	Entradas		Saídas			Suporte a PTO	Suporte a HSC
	120 Vca	24 Vcc/Vca	Relé	Tipo sink de 24 V	Source de 24 V		
2080-LC50-24AWB	14	-	10	-	-	-	-
2080-L50E-24AWB	14	-	10	-	-	-	-
2080-LC50-24QWB	-	14	10	-	-	-	4
2080-L50E-24QWB	-	14	10	-	-	-	4
2080-LC50-24QVB	-	14	-	10	-	2	4
2080-L50E-24QVB	-	14	-	10	-	2	4
2080-LC50-24QBB	-	14	-	-	10	2	4
2080-L50E-24QBB	-	14	-	-	10	2	4
2080-LC50-48AWB	28	-	20	-	-	-	-
2080-L50E-48AWB	28	-	20	-	-	-	-

Tabela 2 - Micro850 Controladores - número e tipos de entradas e saídas (Continuação)

Código de catálogo	Entradas		Saídas			Suporte a PTO	Suporte a HSC
	120 Vca	24 Vcc/Vca	Relé	Tipo sink de 24 V	Source de 24 V		
2080-LC50-48QWB	-	28	20	-	-	-	6
2080-L50E-48QWB	-	28	20	-	-	-	6
2080-LC50-48QWBK	-	28	20	-	-	-	6
2080-L50E-48QWBK	-	28	20	-	-	-	6
2080-LC50-48QVB	-	28	-	20	-	3	6
2080-L50E-48QVB	-	28	-	20	-	3	6
2080-LC50-48QBB	-	28	-	-	20	3	6
2080-L50E-48QBB	-	28	-	-	20	3	6

Tabela 3 - Controladores Micro870 - número e tipos de entradas/saídas

Código de catálogo	Entradas		Saídas			Suporte a PTO	Suporte a HSC
	120 Vca	24 Vcc/Vca	Relé	Tipo sink de 24 V	Source de 24 V		
2080-LC70-24AWB	14	-	10	-	-	-	-
2080-L70E-24AWB	14	-	10	-	-	-	-
2080-LC70-24QWB	-	14	10	-	-	-	4
2080-L70E-24QWB	-	14	10	-	-	-	4
2080-LC70-24QWBK	-	14	10	-	-	-	4
2080-L70E-24QWBK	-	14	10	-	-	-	4
20080-L70E-24QWBN	-	14	10	-	-	-	4
2080-LC70-24QBB	-	14	-	-	10	2	4
2080-L70E-24QBB	-	14	-	-	10	2	4
2080-LC70-24QBBK	-	14	-	-	10	2	4
2080-L70E-24QBBK	-	14	-	-	10	2	4
2080-L70E-24QBBN	-	14	-	-	10	2	4

Cabos de programação

Os controladores Micro800 têm uma interface USB, tornando os cabos USB padrão utilizáveis como cabos de programação.

Use um cabo USB padrão A macho x B macho para programação do controlador.



Cabos para porta serial incorporada

Cabos para porta serial incorporada de comunicação são listados aqui. Todos os cabos para porta serial incorporada precisam ter 3 metros ou menos de comprimento.

Tabela 4 - Tabela de seleção de cabos para porta serial incorporada

Conectores	Comprimento	Cód. de catálogo	Conectores	Comprimento	Cód. de catálogo
Mini DIN de 8 pinos para Mini DIN de 8 pinos	0,5 m (1,5 pés)	1761-CBL-AM00 ⁽¹⁾	Mini DIN de 8 pinos para DB9 de 9 pinos	0,5 m (1,5 pés)	1761-CBL-AP00 ⁽¹⁾
Mini DIN de 8 pinos para Mini DIN de 8 pinos	2 m (6,5 pés)	1761-CBL-HM02 ⁽¹⁾	Mini DIN de 8 pinos para DB9 de 9 pinos	2 m (6,5 pés)	1761-CBL-PM02 ⁽¹⁾
Mini DIN de 8 pinos para Mini DIN de 8 pinos (com mecanismo de trava em ambos os conectores)	2 m (6,5 pés)	1761-CBL-AH02	Mini DIN de 8 pinos com mecanismo de trava para D-shell de 9 pinos	2 m (6,5 pés)	1761-CBL-PH02
—			Bloco de bornes de 6 pinos RS-485 com mini DIN de 8 pinos	30 mm (11,8 pol.)	1763-NC01 série A

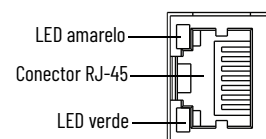
(1) Série C ou posterior para aplicações Classe 1 Div 2.

Suporte Ethernet incorporado

Para controladores Micro850, uma porta base-T 10/100 (com indicadores de LED verdes e amarelos integrados) está disponível para conexão a uma rede Ethernet por meio de qualquer cabo Ethernet RJ-45 padrão. Os indicadores de LED servem como indicadores para status de transmissão e recepção.

Mapeamento de pinos na porta Ethernet RJ-45

Número do contato	Sinal	Direção	Função primária
1	TX+	DESATIVADO	Transmitir dados -
2	TX-	DESATIVADO	Transmitir dados -
3	RX-	IN	Receptor de dados Ethernet diferencial -
4			Terminado
5			Terminado
6	RX+	IN	Receptor de dados Ethernet diferencial +
7			Terminado
8			Terminado
Blindagem			Aterramento do rack



O LED de status amarelo indica Link (amarelo sólido) ou ausência de link (apagado).

O LED de status verde indica atividade (verde intermitente) ou ausência de atividade (apagado).

Os controladores Micro850 e Micro870 suportam cabos cruzados Ethernet (2711P-CBL-EX04).

Indicação de status Ethernet

Controladores Micro850 também suportam dois LEDs para EtherNet/IP para indicar o seguinte:

- Status do módulo
- Status da rede

Consulte [Localização de falhas na página 313](#) para obter descrições dos indicadores de status do módulo e da rede.

Sobre o controlador

Software de programação para controladores Micro800

O software Connected Components Workbench é um conjunto de ferramentas colaborativas compatíveis com os controladores Micro800. É baseado em tecnologias da Rockwell Automation® e Microsoft Visual Studio® e oferece programação do controlador, configuração do dispositivo e integração com editor IHM. Utilize esse software para programar seus controladores, configurar dispositivos e projetar as suas aplicações de interface de operação.

O software Connected Components Workbench oferece uma escolha de linguagens de programação IEC 61131-3 (diagrama de lógica ladder, diagrama de blocos de funções, texto estruturado) compatível com bloco de funções definido pelo usuário que otimiza o controle da máquina.

Obtenha o Software Connected Components Workbench

Está disponível um download gratuito em rok.auto/ccw.

Use o Software Connected Components Workbench

Para ajudá-lo a programar o seu controlador por meio do software Connected Components Workbench, você pode consultar a ajuda online do Connected Components Workbench (que acompanha o software).

IMPORTANTE Os novos controladores Micro850 (2080-L50E) e Micro870 (2080-L70E) são suportados somente a partir da versão do software Connected Components Workbench 20.01.00 em diante.

Alterações do controlador no modo de operação

Os controladores Micro820®/Micro830/Micro850/Micro870 permitem fazer certas alterações durante o modo de operação usando os recursos a seguir:

- Alterar modo de operação (RMC)
Este recurso permite modificações lógicas a um projeto em execução sem ir para o modo de programa remoto.
Para obter mais informações, consulte [Usar alteração do modo de operação \(RMC\) na página 24](#).
- Alteração da configuração do modo de operação (RMCC)|
Este recurso permite alterar a configuração de endereço do controlador a ser feita dentro de um programa durante o modo de operação.
Para obter mais informações, consulte [Usando a alteração do modo de operação \(RMCC\) na página 28](#).

Usar alteração do modo de operação (RMC)

A alteração do modo de operação (RMC) é um recurso de aprimoramento de produtividade suportado pelo software Connected Components Workbench para controladores Micro820/Micro830/Micro850/Micro870. Ele permite ao usuário economizar tempo ao permitir modificações de lógica em um projeto em funcionamento sem ir ao modo de programa remoto e sem se desconectar do controlador.

IMPORTANTE A revisão do firmware do controlador Micro820/Micro830/Micro850 8.xxx ou posterior também é necessária para usar Alterar modo de operação.

RMC é útil quando o usuário está desenvolvendo um projeto pela adição incremental de pequenas alterações na lógica e deseja ver imediatamente os efeitos das alterações na máquina. Com RMC, como o controlador permanece no modo de operação remoto, os atuadores da máquina e de lógica do controlador não precisarão reiniciar constantemente, o que pode ocorrer se o controlador for mudado para o modo de programa (por exemplo, o primeiro bit de varredura é verificado na lógica do programa para liberar saídas).

Quando o usuário está editando, construindo e descarregando um projeto sem usar o RMC, uma construção completa do projeto total do controlador é realizada, além de um download total do projeto. Durante o RMC, uma construção incremental é realizada e somente alterações incrementais são descarregadas para o controlador.

IMPORTANTE Não se desconecte do controlador após realizar a alteração de modo de operação; faça uma construção completa e tente se reconectar. O software Connected Components Workbench trata o projeto no controlador como diferente do projeto no próprio software e pedirá para fazer o upload ou download mesmo que a lógica seja idêntica.

O RMC é realizado de modo incremental no fim de cada varredura de programa para evitar um grande atraso na varredura do programa. Isso adiciona mais 12 ms ao tempo de varredura. Por exemplo, se a varredura do programa leva normalmente 10 ms, ela pode aumentar para 22 ms durante o RMC até que a atualização seja terminada. De modo semelhante, as interrupções do usuário podem ser atrasadas.

Exemplo dos benefícios de usar RMC - 20% de redução no tempo de download

Número de alterações	Tempo para realizar o download convencional (segundos)	Tempo para testar a lógica e aceitar as alterações (segundos)
1	36	29
5	180	130
10	360	255

Tamanho da memória de projeto utilizada para comparação:

Dados = 14784 bytes; Programa = 2352 bytes

Nota: A duração começa quando o botão RMC é clicado enquanto conectado ao controlador e termina quando a aceitação é finalizada. Por exemplo:

1. Quando conectado ao controlador, clique em RMC
 2. Modifique o programa
 3. Clique em Testar lógica
 4. Clique em Aceitar para terminar, ou clique em Testar lógica para fazer outra alteração
-



ATENÇÃO: Use extremo cuidado quando usar Alterar modo de operação. Erros podem ferir o pessoal e danificar equipamentos. Antes de usar Alterar modo de operação:

- avalie como a máquina responderá às mudanças.
- notifique todo o pessoal sobre as alterações.

Uma nova variável global `__SYSVA_PROJ_INCOMPLETE` foi adicionada para indicar quando estão sendo feitas alterações no modo de operação. Isso pode ser utilizado para notificar o pessoal no IHM de que há alterações livres no controlador.

Definições de bits de variável global - `__SYSVA_PROJ_INCOMPLETE`

Bit	Definição
0	Defina quando começa o processo de Alterar modo de operação. Removido depois que Alterar modo de operação é gravado permanentemente no controlador (finalização de Aceitar ou Desfazer). Este bit pode ser usado para avisar os operadores que está em progresso uma alteração do modo de operação e que há alterações livres no controlador.
1	Defina se um erro ocorreu enquanto eram gravadas alterações para atualizar ou uma verificação de integridade falhou durante a alteração do modo de operação. Removido na próxima alteração de modo de operação bem-sucedida.

Quando você realiza uma alteração de lógica de teste, o valor da variável é alterado de zero para um. Após escolher aceitar ou cancelar as alterações, o valor da variável é redefinido para zero.

IMPORTANTE Quando é realizado um Testar lógica, ou ao desfazer as alterações depois que se completa um Testar lógica, quaisquer instruções de comunicação ativas serão abortadas enquanto as alterações são descarregadas para o controlador.

Alterações livres

Alterações não confirmadas são alterações feitas no RMC que não foram aceitas ou foram desfeitas depois que foi realizada uma alteração da lógica de teste.

Se a alimentação do controlador perder potência enquanto houver alterações livres, não será possível entrar novamente no RMC durante uma nova conexão. Você pode escolher fazer um novo download do projeto para manter as alterações, ou carregar se as alterações livres forem indesejadas.

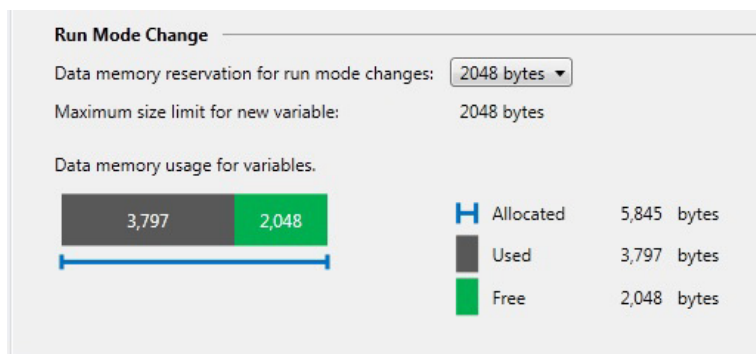
Se você escolher carregar um projeto com alterações livres a partir do controlador, não será possível entrar no RMC até ter realizado um download completo.

Memória RMC

A memória de alteração do modo de operação (RMC) é usada para armazenar as alterações de lógica e de variável do usuário feitas durante o RMC. A quantidade padrão de memória alocada é de 2 KB e pode ser aumentada para até 16 KB. No entanto, ainda há um limite de 2 KB para alterações lógicas e variáveis de usuário por lógica de teste. Para ajustar a quantidade de memória RMC, o controlador deve estar offline. Depois de ter ajustado a quantidade, você deve construir o projeto e descarregá-lo para o controlador.

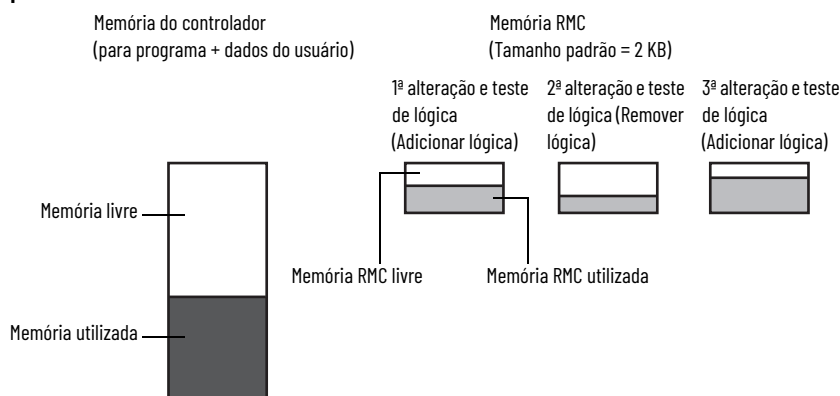
IMPORTANTE Em um projeto do software Connected Components Workbench Release 8, o espaço disponível para dados do usuário foi reduzido para 6 KB para suportar configurações otimizadas de projeto para o novo recurso RMC. Se você tem um projeto que foi desenvolvido antes da versão 8, pode ser necessário reduzir a seção de variáveis temporárias do padrão "Alocado" de 8 KB da página de memória para compilar o projeto com sucesso.

Página de diagnóstico de memória do controlador no software Connected Components Workbench



Durante o RMC, uma construção incremental é realizada e somente alterações incrementais são descarregadas para o controlador até que a memória RMC esteja cheia.

Exemplo de uso da memória RMC

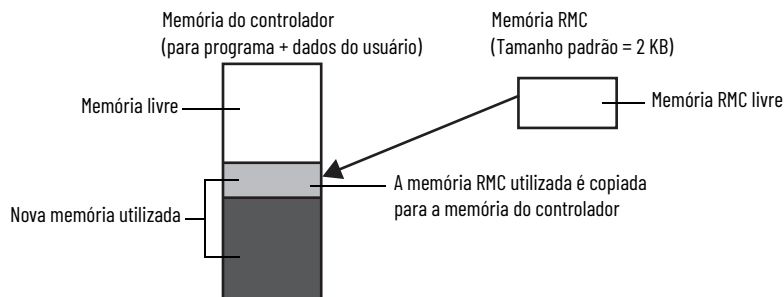


Se não houver memória RMC suficiente disponível para fazer mais alterações (por exemplo, uma mensagem de erro de "não há memória suficiente" aparecer durante a construção de RMC ou lógica de teste), então deve ser realizado um download completo a partir da memória RMC para a memória-padrão de dados e do programa do usuário.

Transferindo conteúdo na memória RMC para a memória do controlador

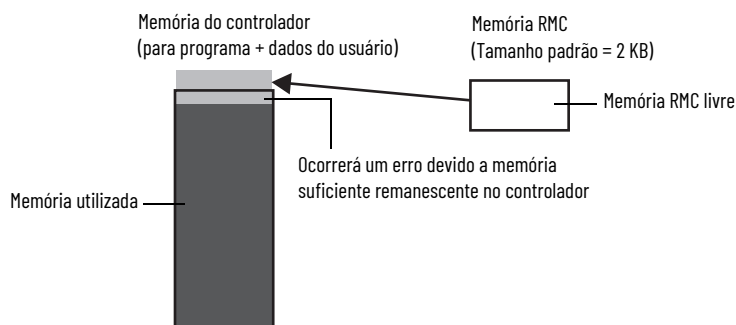
As alterações realizadas durante o RMC são armazenadas na memória RMC e permanecerão ali até que seja realizada uma construção completa e um download (enquanto o controlador está desconectado).

Uso da memória RMC ao executar o exemplo completo de compilação e download



Contudo, se a memória do controlador não tiver espaço remanescente suficiente para copiar o conteúdo da memória RMC como mostrado abaixo, a operação falhará e aparecerá uma mensagem de erro de “não há memória suficiente”. Não use o RMC se a memória do seu controlador estiver próxima dos limites.

Exemplo de memória insuficiente do controlador



Limitações do RMC

Tome nota das seguintes limitações quando usar o recurso de alteração do modo de operação (RMC):

- Não puderam ser realizadas alterações (por exemplo, alterar os tempos de filtro).
- Até 2 KB de variáveis de lógica e de usuário (aproximadamente 150 instruções booleanas⁽¹⁾) e que podem ser adicionadas para cada lógica de teste.
- A memória total alocada para RMC (acumulativa em todas as alterações de lógica de teste) pode ser aumentada de 2 KB para 8 KB, mas permanece o limite de 2 KB por lógica de teste.
- Até 20 POU (unidades organizacionais do programa) podem ser adicionadas para cada alteração (por exemplo, se você tem atualmente 5 POU, pode adicionar mais 20, para um total de 25 POU).

(1) Aproximadamente 85 instruções booleanas para controladores Micro850 (2080-L50E) e Micro870 (2080-L70E).

- Se um bloco de funções definidas pelo usuário (UDFB) é modificado de modo a mudar as variáveis locais, tais variáveis serão reinicializadas ou definidas em zero e será exibida uma mensagem de advertência durante a construção. Se desejar reaplicar o valor inicial, clique com o botão direito no UDFB e selecione Refatoração → Restaurar valores iniciais das instâncias.
- RMC não será possível após realizar uma operação de Descobrir Projeto se um novo módulo for detectado, porque a configuração terá mudado.
- Arquivos trocados não podem ser importados quando no RMC porque isso é considerado uma mudança na configuração.
- A realização de mudanças na configuração da tela (por exemplo, ocultar comentários) é tratada como mudança de lógica e exige que o projeto seja construído.
- Variáveis globais não podem ser apagadas nem modificadas no RMC, mas podem ser adicionadas. Para excluir ou modificar uma variável global, o software Connected Components Workbench deve ser desconectado do controlador.
- Quando utilizar mensagens CIP no RMC, não surte efeito definir o parâmetro de tipo de dados CIPTARGETCFG ConnClose para TRUE. A sessão Ethernet não se fecha imediatamente após uma mensagem bem-sucedida e você deve esperar pelo tempo-limite da conexão após 60 segundos. Isso se aplica a projetos do software Connected Components Workbench versão 9 ou anteriores. Para projetos da versão 10 ou posteriores, é configurável o tempo limite da conexão CIP.



ADVERTÊNCIA: Se você excluir a linha de programa de saída quando em Alterar modo de operação e aceitar as alterações, a saída no controlador permanecerá ligada.

Consulte [Usar alteração do modo de operação na página 291](#) para ver um exemplo de como usar este recurso.

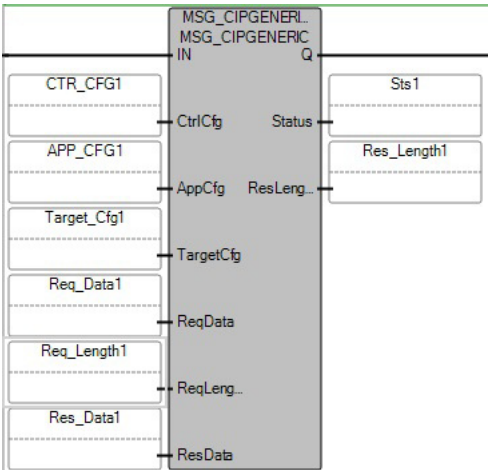
Usando a alteração do modo de operação (RMCC)

A alteração da configuração do modo de operação (RMCC) é um recurso de aprimoramento de produtividade suportado pelo software Connected Components Workbench para controladores Micro820/Micro830/Micro850/Micro870. Permite que os usuários reutilizem um programa idêntico com vários controladores simplesmente modificando a configuração do endereço de um controlador dentro do programa durante o modo de operação. A revisão do firmware do controlador Micro820/Micro830/Micro850 9.xxx ou posterior é necessária para usar esse recurso.

RMCC pode ser utilizado para modificar a configuração do endereço do controlador durante o modo de operação quando o protocolo de comunicação está definido para Modbus RTU para portas seriais ou EtherNet/IP para a porta Ethernet. RMCC usa uma mensagem CIP genérica que somente pode ser enviada de dentro de um programa de controlador e não a partir de um dispositivo externo ao controlador.

IMPORTANTE Durante o RMCC, o tempo de varredura pode aumentar para perto de 100 ms. Não execute o RMCC se o controlador estiver realizando operações críticas de tempo.

Instrução de mensagem CIP genérica para alteração da configuração do modo de operação



A alteração da configuração do modo de operação (RMCC) somente pode ser realizada pelo controlador que está enviando a mensagem. Para fazer isso, é necessário configurar a mensagem CIP genérica como uma mensagem em malha reversa definindo o caminho para “o,o”.

Configure a mensagem CIP genérica como uma mensagem em malha reversa

Name	Data Type	Dimension	String Size	Initial
Target_Cfg1	CIPTARGETCFG			...
Target_Cfg1.Path	STRING		80	'0,0'
Target_Cfg1.CipConnMode	USINT			0
Target_Cfg1.UcmmTimeout	UDINT			0
Target_Cfg1.ConnMsgTimeout	UDINT			0
Target_Cfg1.ConnClose	BOOL			

Para controladores Micro830/Micro850/Micro870, a mudança de configuração do endereço é permanente e será mantida quando o controlador for desligado e ligado novamente. A partir da revisão de firmware 10 em diante, os controladores Micro820 também mantêm a configuração de endereço quando o controlador é ligado em ciclo.

Uso da comunicação Modbus RTU

Para usar RMCC com o protocolo de comunicação Modbus RTU, a porta serial deve ser definida para o papel de slave Modbus. Uma mensagem CIP genérica é enviada de dentro de um programa com os seguintes parâmetros.

Parâmetros de mensagem genérica CIP para RMCC usando Modbus RTU

Parâmetro	Valor
Serviço	16
Classe	70
Instância	2 - Porta serial incorporada 5, 6, 7, 8, ou 9 - Módulos plug-in
Atributo	100
ReqData	Novo endereço do nó, 1
ReqLen	2

Exemplo de Modbus RMCC - Definir os parâmetros

User Global Variables - Micro850					
Local Variables - RMCC_Modbus					
Name	Data Type	Dimension	String Size	Initial Value	
APP_CFG1	CIPAPPCFG				Rea
APP_CFG1.Service	USINT			16	Rea
APP_CFG1.Class	UINT			70	Rea
APP_CFG1.Instance	UDINT			2	Rea
APP_CFG1.Attribute	UINT			100	Rea
APP_CFG1.MemberCnt	USINT				Rea
APP_CFG1.MemberId	CIPMEMBERID				Rea

Exemplo de Modbus RMCC - Definir o endereço do novo nó

User Global Variables - Micro850					
Local Variables - RMCC_Modbus					
Name	Data Type	Dimension	String Size	Initial Value	
Req_Data1	USINT	[1..70]			
Req_Data1[1]	USINT			3	
Req_Data1[2]	USINT			1	

O primeiro byte indica o novo endereço do nó para o controlador. Para este exemplo, o endereço do nó é “3”. O segundo byte sempre deve ser “1”, isso indica que o papel do Modbus está configurado como Slave.

Exemplo de Modbus RMCC - Definir comprimento da mensagem

User Global Variables - Micro850					
Local Variables - RMCC_Modbus					
Name	Data Type	Dimension	String Size	Initial Value	
Req_Length1	UINT			2	Re
Res_Length1	UINT				Re

Quando o novo endereço do nó é configurado e aplicado, a porta não é reiniciada.

IMPORTANTE Você deve garantir que o nodo endereço do nó que está sendo configurado é único, pois ele não será verificado em relação aos endereços de nós existentes em outros dispositivos.

Você pode verificar se o endereço do nó mudou depois de realizar RMCC olhando a aba Communication Diagnostics do controlador.

Exemplo de Modbus RMCC - Verificar alteração do endereço

Micro850

Micro850 - Communication Diagnostics

Communication: Serial Port

Reset Counters

Channel: Slot 1 | 2080-SERIALISOL at port 5

Drivers: Modbus RTU

Link Counters

Characters Received:	3,088	Characters Sent:	2,464
Frame Received:	386	Frames Sent:	352
Good Transactions:	352	Broadcasts:	0
Good Exceptions:	0	Mismatch Errors:	0
Bad CRC:	0	No Response:	0
		Other Errors:	0

Common Settings

Unit Address: 3

Usando comunicação EtherNet/IP

Para usar RMCC com o protocolo de comunicação EtherNet/IP, o controlador deve estar configurado para usar um endereço IP estático. Se o controlador estiver configurado para usar BOOTP ou DHCP, a alteração será rejeitada. Uma mensagem CIP genérica é enviada de dentro de um programa com os seguintes parâmetros.

Use o RMCC ao configurar o controlador durante o comissionamento. Imediatamente após alterar o endereço IP, o tempo de ciclo pode aumentar até 100 ms para uma digitalização de programa.

Parâmetros de mensagem genérica CIP para RMCC usando EtherNet/IP

Parâmetro	Valor
Serviço	16
Classe	245
Instância	1
Atributo	5
ReqData	Endereço IP, máscara de sub-rede, endereço do conversor de protocolos
ReqLen	22 bytes

Exemplo de EtherNet/IP RMCC – Definir os parâmetros

Name	Data Type	Dimension	String Size	Initial Value	Attrib
APP_CFG1	CIPAPPCFG				Read/Wr
APP_CFG1.Service	USINT			16	Read/Wr
APP_CFG1.Class	UINT			245	Read/Wr
APP_CFG1.Instance	UDINT			1	Read/Wr
APP_CFG1.Attribute	UINT			5	Read/Wr
APP_CFG1.MemberCnt	USINT				Read/Wr
APP_CFG1.MemberId	CIPMEMBERID				Read/Wr

Exemplo de EtherNet/IP RMCC – Definir novo endereço IP

Name	Data Type	Dimension	String Size	Initial Value
Req_Data1	USINT	[1..20]		
Req_Data1[1]	USINT			10
Req_Data1[2]	USINT			1
Req_Data1[3]	USINT			168
Req_Data1[4]	USINT			192
Req_Data1[5]	USINT			0
Req_Data1[6]	USINT			255
Req_Data1[7]	USINT			255
Req_Data1[8]	USINT			255
Req_Data1[9]	USINT			1
Req_Data1[10]	USINT			1
Req_Data1[11]	USINT			168
Req_Data1[12]	USINT			192
Req_Data1[13]	USINT			0
Req_Data1[14]	USINT			0
Req_Data1[15]	USINT			0
Req_Data1[16]	USINT			0
Req_Data1[17]	USINT			0
Req_Data1[18]	USINT			0
Req_Data1[19]	USINT			0
Req_Data1[20]	USINT			0
Req_Data1[21]	USINT			0

Para este exemplo, o novo endereço IP está definido como o seguinte:

- Endereço IP = 192.168.1.10
- Máscara de sub-rede 255.255.255.0
- Endereço do conversor de protocolos = 192.168.1.1

Exemplo de EtherNet/IP RMCC - Definir comprimento da mensagem

User Global Variables - Micro850		Local Variables - RMCC_EIP_LengthTest_SimplePro		System Variables - Micro850	
Name	Data Type	Dimension	String Size	Initial Value	
Req_Length1	UINT			22	Re
Res_Length1	UINT				Re

Depois que o endereço IP for configurado e aplicado, o controlador se desconectará do software Connected Components Workbench se a comunicação for feita por Ethernet.

IMPORTANTE Controladores Micro830 não suportam alterações da configuração do modo de operação usando EtherNet/IP.

IMPORTANTE Você não deve realizar mudanças de endereço IP continuamente. Deixe um intervalo de pelo menos seis segundos antes de realizar a próxima mudança de endereço IP para que a detecção de endereços duplicados funcione adequadamente.

Você pode verificar se o endereço IP foi alterado após a execução do RMCC consultando as configurações de Ethernet para o controlador.

Exemplo de EtherNet/IP RMCC - Verificar alteração do endereço

Controller - Ethernet

Internet Protocol (IP) Settings

☐ Obtain IP address automatically using DHCP

☒ Configure IP address and settings

IP Address: 192 . 168 . 1 . 10

Subnet Mask: 255 . 255 . 255 . 0

Gateway Address: 192 . 168 . 1 . 1

☒ Detect duplicate IP address

Port Settings

Port State: ☒ Enabled ☐ Disabled

☒ Auto-Negotiate Speed and Duplex Mode

Considerações de segurança

Considerações de segurança são um elemento importante da instalação adequada do sistema. Pensar ativamente sobre a sua própria segurança e a dos outros, assim como as condições do seu equipamento, é de importância primária. Recomendamos que você revise as considerações de segurança a seguir.

Desconectar a alimentação principal

A chave seccionadora principal deve estar localizada onde os operadores e pessoal de manutenção tenham acesso rápido e fácil a ela. Além de desligar a alimentação elétrica, todas as outras fontes de energia (pneumática e hidráulica) devem ser desenergizadas antes de trabalhar com uma máquina ou processo comandada por um controlador.

**ADVERTÊNCIA:** Risco de explosão

Não substitua componentes, não conecte ou desconecte o equipamento a não ser que a alimentação tenha sido desligada.

Circuitos de segurança

Circuitos instalados na máquina por razões de segurança, como chaves de sobrecurso, botões de parada e intertravamentos devem sempre ser ligados por cabo diretamente ao relé de controle mestre. Esses dispositivos precisam ser ligados por cabo em série de modo que quando qualquer dispositivo ficar aberto, o relé de controle mestre será desenergizado, cortando portanto a alimentação para a máquina. Nunca altere esses circuitos para anular sua função. Ferimentos graves ou danos a máquina podem ocorrer como resultado.

**ADVERTÊNCIA:** Risco de explosão

Não conecte ou desconecte conectores enquanto o circuito estiver ligado.

Distribuição de potência

Existem alguns pontos sobre distribuição de potência que você deve conhecer:

- O relé de controle mestre precisa ser capaz de inibir todo o movimento da máquina ao cortar a alimentação aos dispositivos de E/S desta quando o relé é desenergizado. É recomendável que o controlador permaneça com alimentação mesmo quando o relé de controle mestre estiver desenergizado.
- Se você estiver utilizando uma fonte de alimentação CC, interrompa o lado da carga em vez da alimentação CA. Isso evita o atraso adicional no desligamento da fonte de alimentação. A fonte de alimentação CC deve ser alimentada diretamente a partir do secundário com fusível do transformador. Alimentação para os circuitos de entrada e saída CC deve ser ligada por meio de um conjunto de contatos do relé de controle mestre.

Testes periódicos do Circuito do relé de controle mestre

Qualquer peça pode falhar, incluindo as chaves em um circuito de relé de controle mestre. A falha de uma dessas chaves muito provavelmente causaria um circuito aberto, o que seria uma falha com corte seguro da alimentação. Porém, se uma dessas chaves entrar em curto-circuito, ela não fornecerá mais nenhuma proteção de segurança. Essas chaves devem ser testadas periodicamente para garantir que irão parar o movimento da máquina quando necessário.

Considerações sobre alimentação

A explicação a seguir expõe considerações sobre alimentação para os microcontroladores.

Transformadores de isolamento

Você pode desejar utilizar um transformador de isolamento na linha de alimentação CA para o controlador. Esse tipo de transformador fornece isolamento do seu sistema de distribuição de potência para reduzir o ruído elétrico que adentra o controlador e é utilizado frequentemente como um transformador de redução da tensão de linha. Qualquer transformador utilizado com o controlador precisa possuir uma potência nominal suficiente para sua carga. A potência nominal é expressa em volt-ampères (VA).

Energização da fonte de alimentação

Durante a energização, a Micro800 fonte de alimentação permite uma breve corrente de energização para carregar os capacitores internos. Muitas linhas de alimentação e transformadores de controle podem fornecer corrente de energização por um período breve. Se a fonte de alimentação não pode fornecer essa corrente de energização, a tensão de fornecimento pode cair momentaneamente.

O único efeito de corrente de energização limitada e queda de tensão no Micro800 é que os capacitores da fonte de alimentação carregam mais lentamente. Porém, o efeito de uma queda de tensão em outros equipamentos deve ser considerado. Por exemplo, uma queda de tensão acentuada pode reinicializar um computador conectado à mesma fonte de alimentação. As considerações a seguir determinam se é necessário ou não que a fonte de alimentação forneça uma corrente de energização elevada:

- A sequência de energização de dispositivos em um sistema.
- A quantidade de queda de tensão na fonte de alimentação caso a corrente de energização não possa ser fornecida.
- O efeito da queda de tensão em outros equipamentos do sistema.

Se você ligar a alimentação ao sistema todo simultaneamente, uma breve queda na tensão da fonte de alimentação normalmente não afetará nenhum equipamento.

Perda da fonte de alimentação

A fonte de alimentação opcional Micro800 CA é projetada para suportar perdas de potência breves sem afetar a operação do sistema. O tempo pelo qual o sistema permanece operacional durante a perda de potência chama-se tempo de sobrevida da varredura do programa após a perda de potência. A duração do tempo de sobrevida da fonte de alimentação depende do consumo de energia do sistema do controlador, mas geralmente fica entre 10 milissegundos e 3 segundos.

Estados de entrada ao cortar a alimentação

O tempo de sobrevida da fonte de alimentação, como descrito acima, é geralmente mais longo que os tempos de início e desligamento das entradas. Por isso, a mudança de estado da entrada de “Ativada” para “Desativada” que ocorre quando a alimentação é cortada pode ser registrada pelo processador antes que a fonte de alimentação desligue o sistema. É importante entender bem esse conceito. O programa do usuário deve ser escrito para levar esse efeito em consideração.

Outros tipos de condições da linha

Ocasionalmente, a fonte de alimentação para o sistema pode ser temporariamente interrompida. Também é possível que o nível de tensão caia substancialmente abaixo da faixa de tensão normal por um período de tempo. Ambas essas condições são consideradas como sendo uma perda de potência para o sistema.

Prevenindo aquecimento excessivo

Para a maioria das aplicações, resfriamento por convecção normal mantém o controlador dentro da faixa de operação especificada. Certifique-se de que a faixa de temperatura especificada seja mantida. Espaçamento apropriado entre os componentes dentro de um gabinete geralmente é suficiente para dissipação de calor.

Em algumas aplicações, uma quantia substancial de calor é produzida por outros equipamentos dentro ou fora do gabinete. Nesse caso, posicione ventiladores sopradores dentro do gabinete para auxiliar a circulação de ar e reduzir sobreaquecimentos localizados próximos ao controlador.

Medidas de resfriamento adicionais podem ser necessárias quando temperaturas ambiente elevadas forem encontradas.



Não permita a entrada de ar externo não filtrado. Posicione o controlador em um gabinete para protegê-lo de uma atmosfera corrosiva. Contaminantes nocivos ou sujeira podem causar operação imprópria ou danificar os componentes. Em casos extremos, você pode precisar utilizar ar condicionado para proteção contra acúmulo de calor dentro do gabinete.

Relé de controle mestre

Um relé de controle mestre (MCR) ligado por cabo fornece um meio confiável para desligamento de emergência da máquina. Já que o relé de controle mestre permite o posicionamento de diversas chaves de parada de emergência em locais diferentes, a sua instalação é importante de um ponto de vista de segurança. Chaves de sobrecurso ou botões de pressão cabeça-de-cogumelo são ligados em série de modo que quando qualquer um deles for aberto, secciona-se a alimentação ao relé de controle mestre. Isso corta a alimentação para os circuitos do dispositivo de entrada e saída. Consulte a [Figura 1 na página 37](#) e a [Figura 2 na página 38](#).



ADVERTÊNCIA: Nunca altere esses circuitos para anular sua função, isso pode resultar em ferimentos graves e/ou danos à máquina.



Se você estiver utilizando uma fonte de alimentação CC externa, interrompa o lado da saída CC em vez do lado da linha CA da fonte para evitar o atraso adicional no desligamento da fonte de alimentação.

A linha CA da fonte de alimentação de saída CC deve ser protegida com fusível.

Conecte um conjunto de relés de controle mestres em série com a potência CC alimentando os circuitos de entrada e saída.

Posicione a chave seccionadora principal da alimentação onde os operadores e pessoal de manutenção tenham acesso rápido e fácil a ela. Se você montar uma chave seccionadora dentro do gabinete do controlador, posicione a manopla da chave no lado externo do gabinete, de modo que você possa cortar a alimentação sem abrir o gabinete.

Sempre que qualquer uma das chaves de parada de emergência for aberta, a alimentação para os dispositivos de entrada e saída deverá ser cortada.

Quando você utiliza o relé de controle mestre para remover a energia dos circuitos de E/S externos, a energia continua a ser fornecida para a fonte de alimentação do controlador, de modo que os indicadores de diagnóstico no processador ainda podem ser observados.

O relé de controle mestre não é um substituto para uma seccionadora da alimentação ao controlador. Ele é destinado para qualquer situação em que o operador precise desenergizar rapidamente somente dispositivos de E/S. Ao inspecionar ou instalar conexões terminais, substituir fusíveis de saída ou trabalhar em equipamento localizado dentro do gabinete, utilize a seccionadora para cortar a alimentação ao resto do sistema.



Não comande o relé de controle mestre com o controlador. Ofereça ao operador a segurança de uma conexão direta entre uma chave de parada de emergência e o relé de controle mestre.

Utilizando chaves de parada de emergência

Ao utilizar chaves de parada de emergência, atenha-se aos pontos a seguir:

- Não programe chaves de parada de emergência no programa do controlador. Qualquer chave de parada de emergência deve desligar toda a alimentação à máquina pelo desligamento do relé de controle mestre.
- Observe todos os códigos locais aplicáveis com relação ao posicionamento e identificação de chaves de parada de emergência.
- Instale as chaves de parada de emergência e o relé de controle mestre no seu sistema. Certifique-se de que contatos de relé possuam uma classificação suficiente para a sua aplicação. Chaves de parada de emergência precisam ficar a fácil alcance.
- Na ilustração a seguir, são demonstrados circuitos de entrada e saída com proteção MCR. Porém, na maioria das aplicações, somente circuitos de saída exigem proteção MCR.

As ilustrações a seguir demonstram o relé de controle mestre ligado a um sistema aterrado.



Na maioria das aplicações, circuitos de entrada não exigem proteção MCR; porém, se você precisar cortar a alimentação a todos os dispositivos de campo, você precisa incluir contatos MCR em série com a cablagem da alimentação de entrada.

Figura 1 - Esquema - Utilizando símbolos IEC

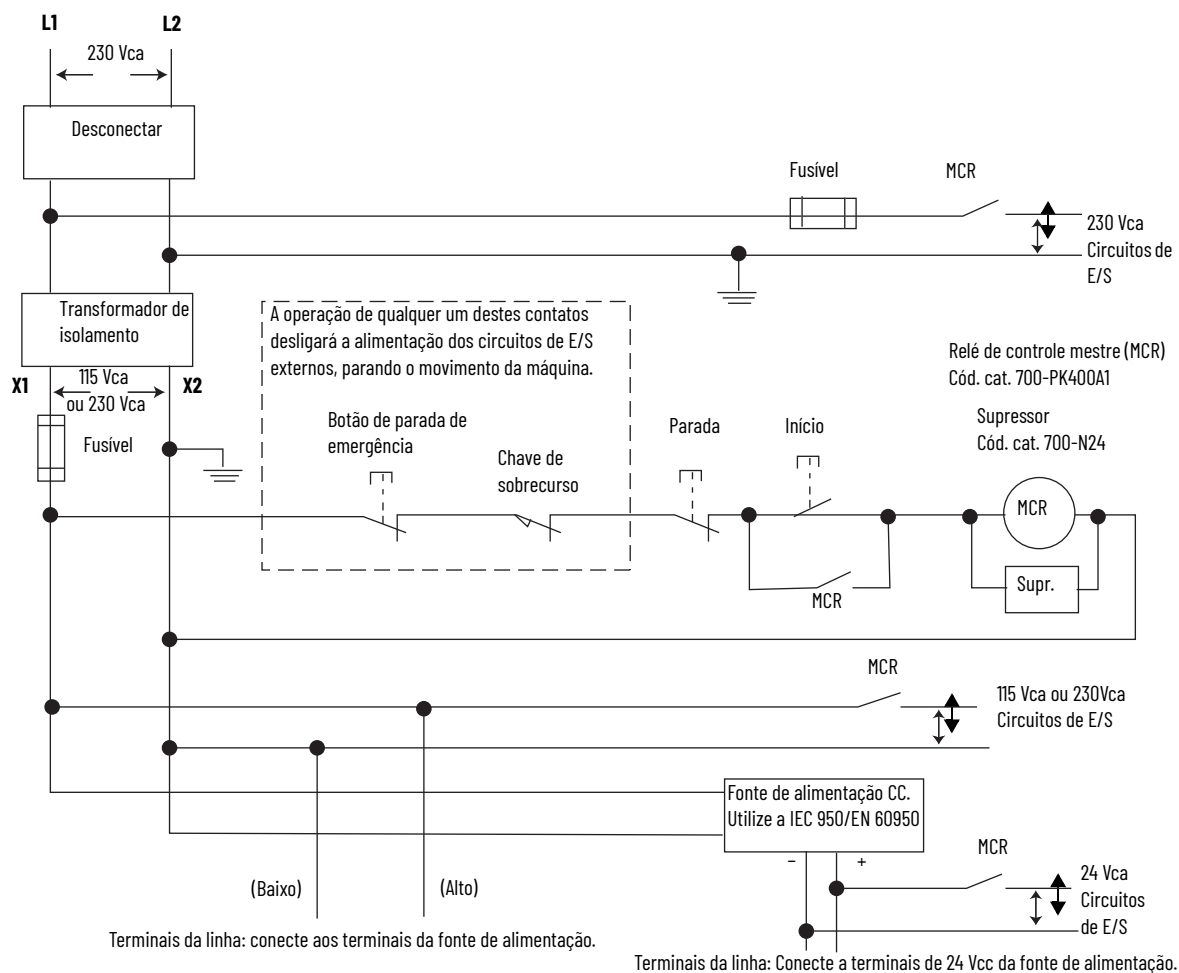
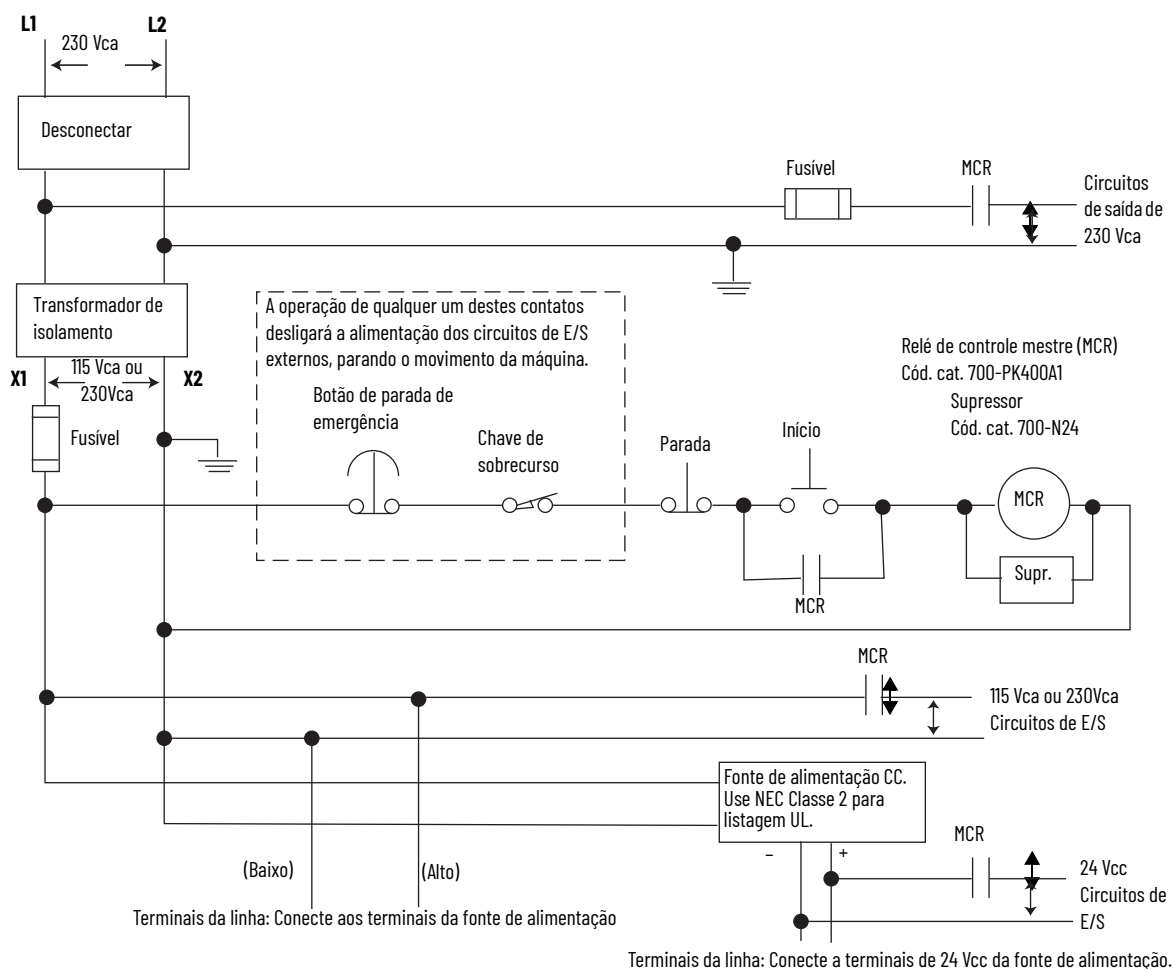


Figura 2 - Esquema - Utilizando símbolos padrão ANSI/CSA



Instale seu controlador

Este capítulo serve como guia para o usuário na instalação do controlador. Ele inclui os tópicos a seguir.

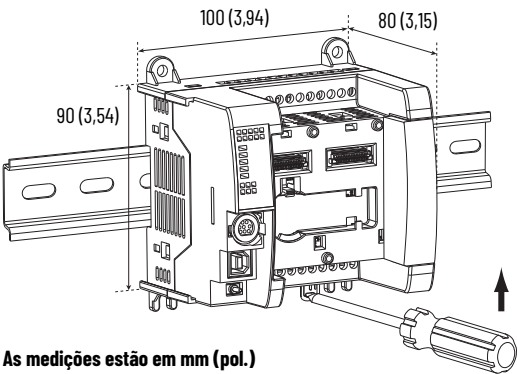
Tópico	Página
Dimensões de montagem do controlador	39
Dimensões de montagem	39
Montagem em trilho DIN	41
Montagem em painel	41

Dimensões de montagem do controlador

Dimensões de montagem

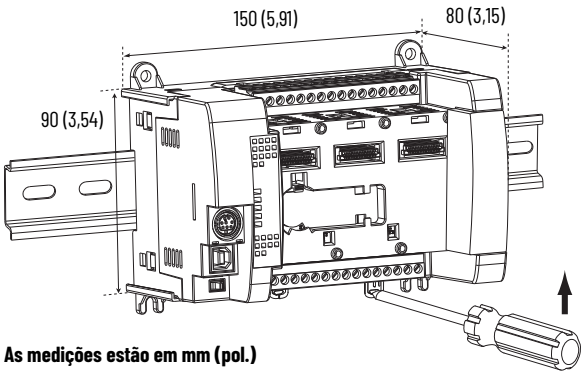
As dimensões de montagem não incluem pés de montagem nem travas de trilho DIN.

Micro830 Controladores de 10 e 16 pontos
2080-LC30-10QWB, 2080-LC30-10QVB, 2080-LC30-16AWB, 2080-LC30-16QWB, 2080-LC30-16QVB



As medições estão em mm (pol.)

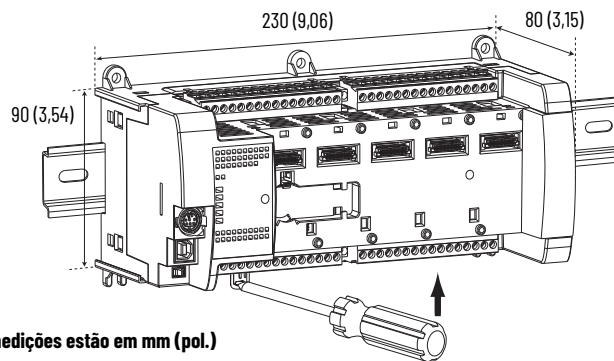
Micro830 Controladores de 24 pontos
2080-LC30-24QW8B, 2080-LC30-24QVB, 2080-LC30-24QBB



As medições estão em mm (pol.)

Micro830 Controladores de 48 pontos

2080-LC30-48AWB, 2080-LC30-48QWB, 2080-LC30-48QVB, 2080-LC30-48QBB



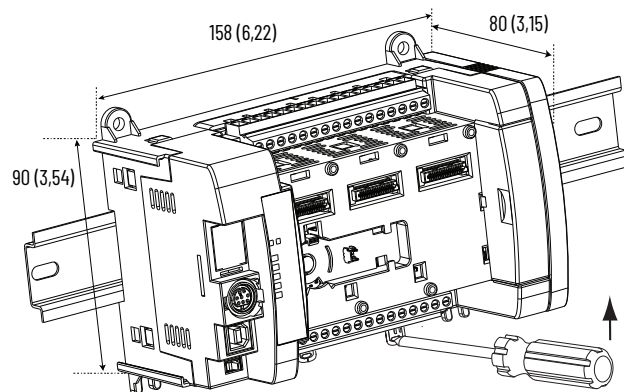
As medições estão em mm (pol.)

Micro850 Controladores de 24 pontos

2080-LC50-24AWB, 2080-LC50-24QBB, 2080-LC50-24QVB, 2080-LC50-24QWB, 2080-L50E-24AWB, 2080-L50E-24QBB, 2080-L50E-24QVB, 2080-L50E-24QWB

Controladores Micro870 de 24 pontos

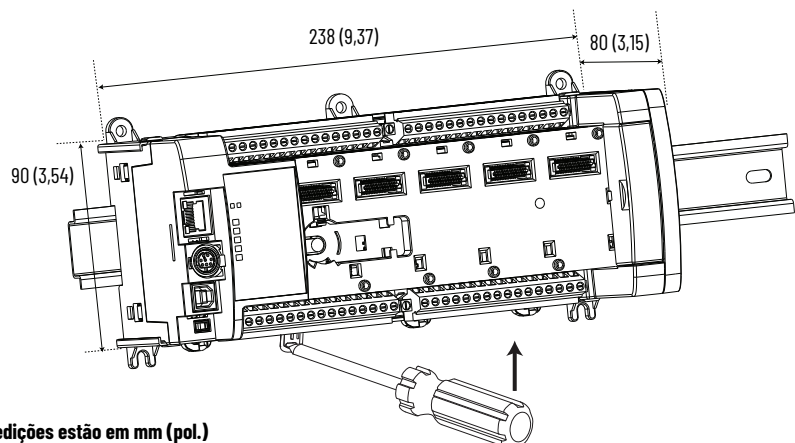
2080-LC70-24AWB, 2080-LC70-24QWB, 2080-LC70-24QWBK, 2080-LC70-24QBB, 2080-LC70-24QBBK, 2080-L70E-24AWB, 2080-L70E-24QWB, 2080-L70E-24QWBK, 2080-L70E-24QWBN, 2080-L70E-24QBB, 2080-L70E-24QBBK, 2080-L70E-24QBBN



As medições estão em mm (pol.)

Micro850 Controladores de 48 pontos

2080-LC50-48AWB, 2080-LC50-48QWB, 2080-LC50-48QWBK, 2080-LC50-48QBB, 2080-LC50-48QVB, 2080-L50E-48AWB, 2080-L50E-48QWB, 2080-L50E-48QWBK, 2080-L50E-48QBB, 2080-L50E-48QVB



As medições estão em mm (pol.)

Mantenha-o longe de objetos como paredes do gabinete, condutores e equipamentos adjacentes. Deixe um espaço de 50,8 mm (2 pol.) em todos os lados para obter uma ventilação adequada. Se forem conectados acessórios/módulos opcionais ao controlador, como a fonte de alimentação 2080-PS120-240 VAC ou módulos de expansão de E/S, certifique-se de que haja um espaço de 50,8 mm (2 pol.) em todos os lados depois de conectar as peças opcionais.

Montagem em trilho DIN

O módulo pode ser montado usando os seguintes trilhos DIN: 35 x 7,5 mm x 1 mm (EN 50 022 - 35 x 7,5).



Para ambientes com maiores preocupações de vibração e choque, use o método de montagem do painel, em vez de montagem em trilho DIN.

Antes de montar o módulo em um trilho DIN, use uma chave de fenda na trava do trilho DIN e faça um movimento de alavanca para baixo até que esteja na posição destravada.

1. Enganche a parte superior da área de montagem do trilho DIN do controlador no próprio trilho DIN e pressione a parte inferior até que o controlador se encaixe no trilho DIN.
2. Empurre a trava do trilho DIN de volta para o ponto de retenção. Utilize as fixações de terminação dos trilhos DIN (número de peça Allen-Bradley 1492-EAJ35 ou 1492-EAHJ35) para ambientes de vibração ou choque.

Para remover seu controlador do trilho DIN, faça uma alavanca para baixo até que uma trava do trilho DIN esteja na posição destravada.

Montagem em painel

O método de montagem recomendado é usar quatro parafusos M4 (nº 8) por módulo. Tolerância do espaçamento do furo: $\pm 0,4$ mm (0,016 pol.).

Siga essas etapas para instalar seu controlador usando os parafusos de fixação.

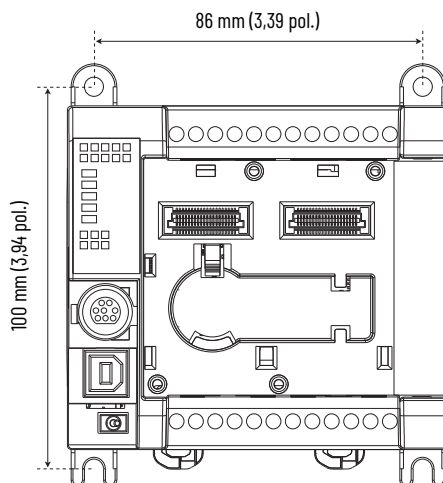
1. Coloque o controlador contra o painel em que ele está sendo montado. Certifique-se de que o controlador tenha o espaçamento correto.
2. Marque os locais dos furos através dos furos dos parafusos de montagem e os pés de montagem, em seguida, remova o controlador.
3. Faça os orifícios nas indicações, substitua o controlador e instale-o. Deixe a proteção contra detritos no lugar até terminar a fiação do controlador e de todos os outros dispositivos.

IMPORTANTE Para obter instruções sobre como instalar o sistema Micro800 com expansão de E/S, consulte Módulos de expansão do Micro800, Manual do usuário, publicação [2080-UM003](#).

Dimensões da montagem em painel

Micro830 Controladores de 10 e 16 pontos

2080-LC30-10QWB, 2080-LC30-10QVB, 2080-LC30-16AWB, 2080-LC30-16QWB, 2080-LC30-16QVB



Micro830 Controladores de 24 pontos

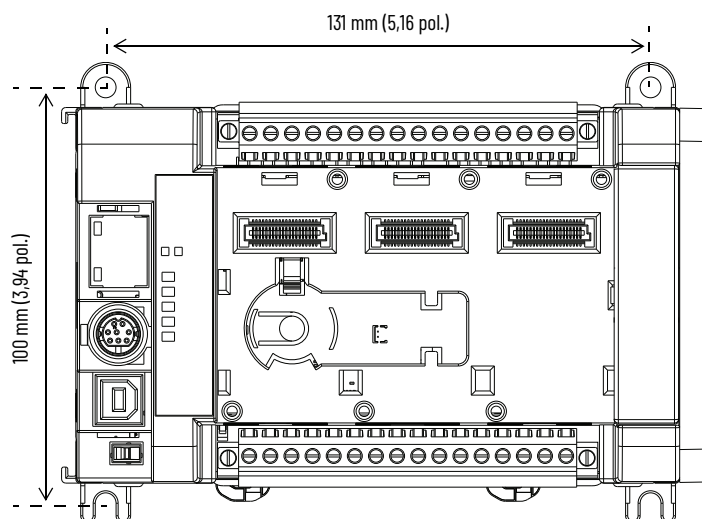
2080-LC30-24QWB, 2080-LC30-24QVB, 2080-LC30-24QBB

Micro850 Controladores de 24 pontos

2080-LC50-24AWB, 2080-LC50-24QBB, 2080-LC50-24QVB, 2080-LC50-24QWB, 2080-L50E-24AWB, 2080-L50E-24QBB, 2080-L50E-24QVB, 2080-L50E-24QWB

Controladores Micro870 de 24 pontos

2080-LC70-24AWB, 2080-LC70-24QWB, 2080-LC70-24QWBK, 2080-LC70-24QBB, 2080-LC70-24QBBK, 2080-L70E-24AWB, 2080-L70E-24QWB, 2080-L70E-24QWBK, 2080-L70E-24QWBN, 2080-L70E-24QBB, 2080-L70E-24QBBK, 2080-L70E-24QBBN

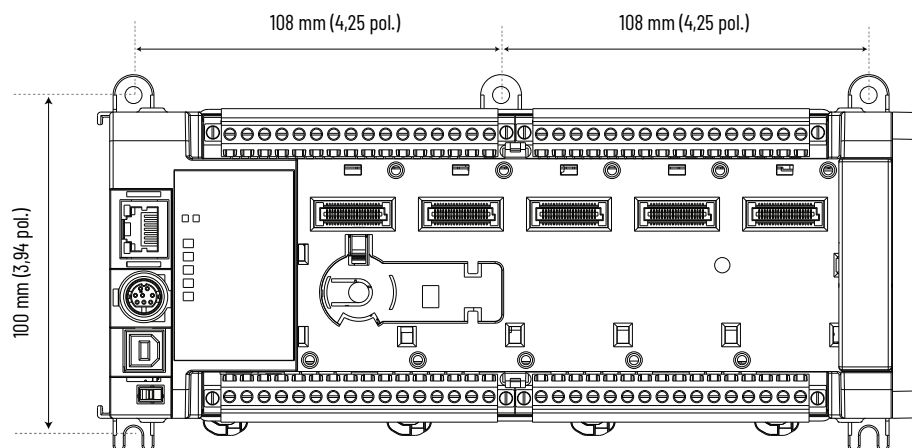


Micro830 Controladores de 48 pontos

2080-LC30-48AWB, 2080-LC30-48QWB, 2080-LC30-48QVB, 2080-LC30-48QBB

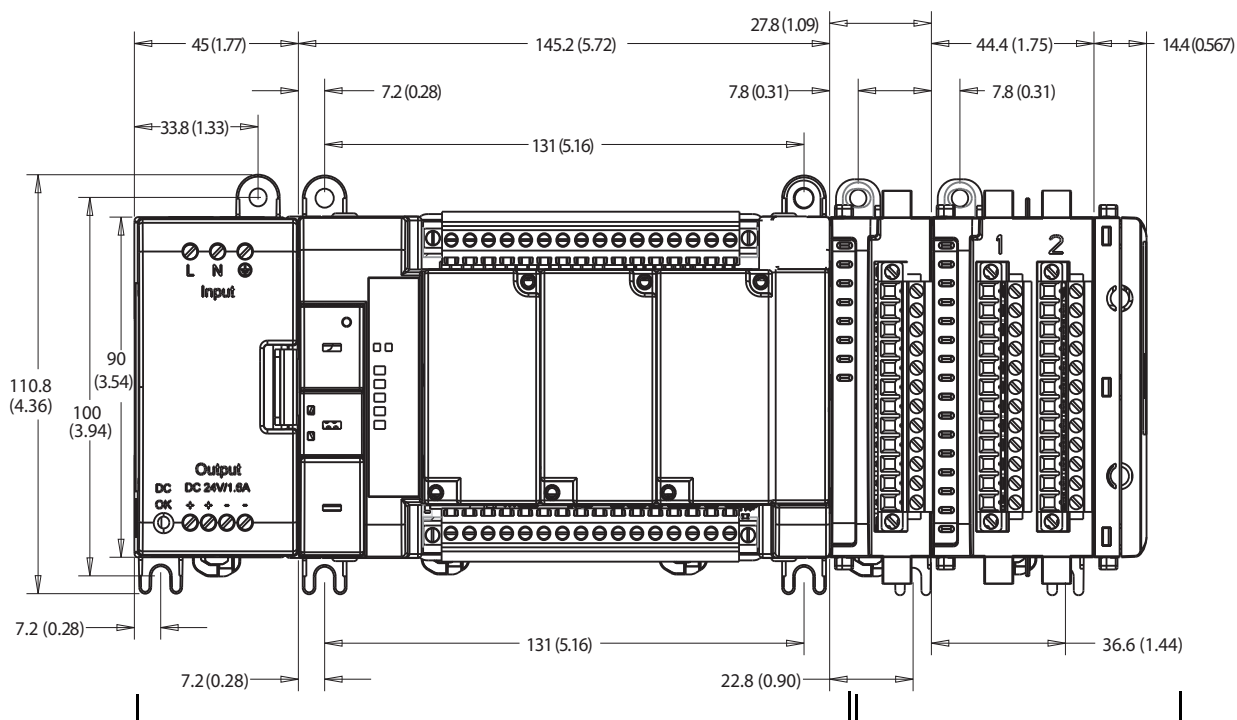
Micro850 Controladores de 48 pontos

2080-LC50-48AWB, 2080-LC50-48QWB, 2080-LC50-48QWBK, 2080-LC50-48QBB, 2080-LC50-48QVB, 2080-L50E-48AWB, 2080-L50E-48QWB, 2080-L50E-48QWBK, 2080-L50E-48QBB, 2080-L50E-48QVB



Montagem do sistema

Controladores de 24 pontos Micro830, Micro850 e Micro870 (Frontal)



Controlador Micro830/Micro850/Micro870 de 24 pontos com fonte de alimentação Micro800

Slots de expansão de E/S

(aplicável somente a Micro850 e Micro870)

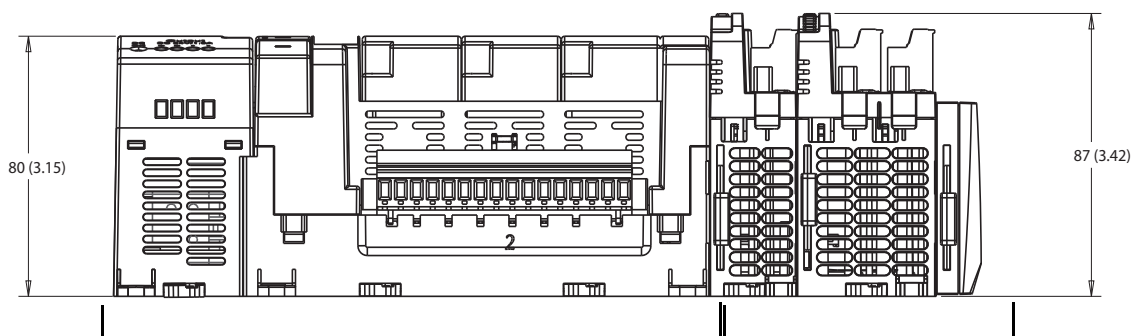
Largura única (1º slot)

Largura dupla (2º slot)

2085-ECR (terminador)

As medições estão em mm (pol.)

Controladores de 24 pontos Micro830, Micro850 e Micro870 (Lateral)

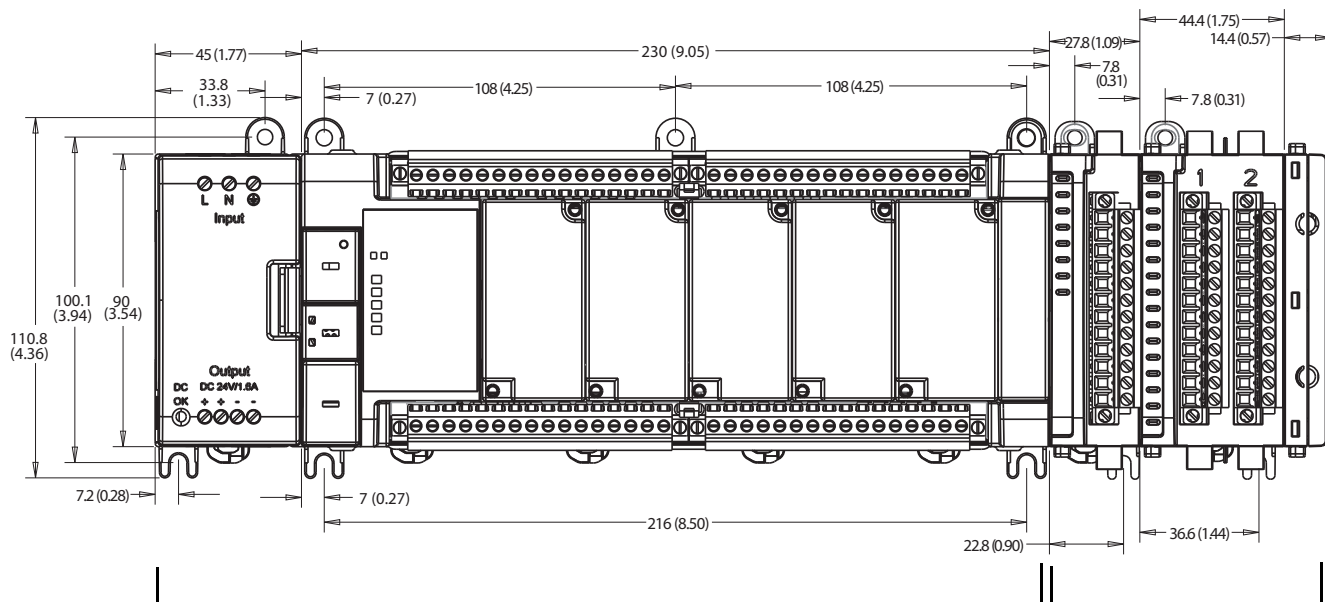


Controlador Micro830/Micro850/Micro870 de 24 pontos com fonte de alimentação Micro800

Slots de expansão de E/S
(aplicável somente a Micro850 e Micro870)
Largura única (1º slot)
Largura dupla (2º slot)
2085-ECR (terminador)

As medições estão em mm (pol.)

Controladores de 48 pontos Micro830 e Micro850 (Frontal)

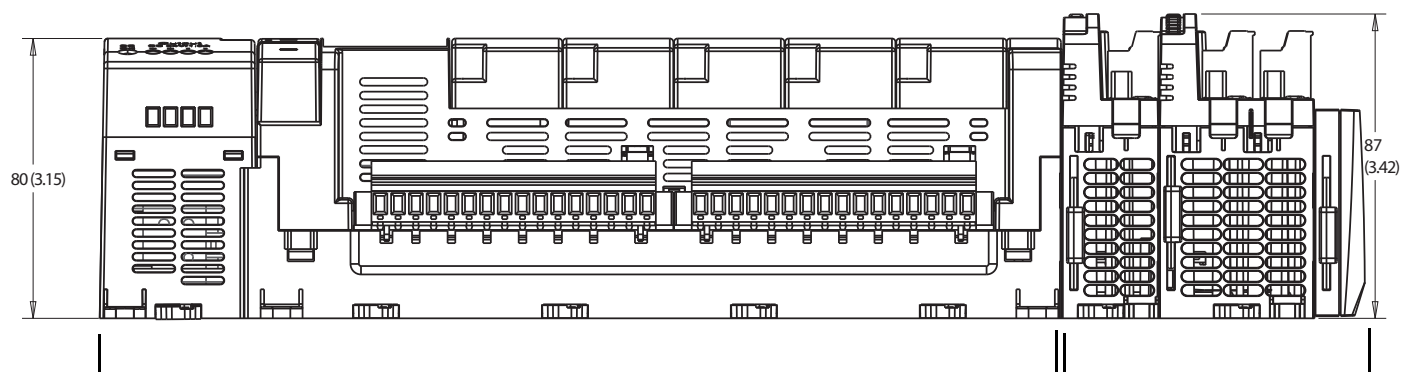


Controlador Micro830/Micro850 de 48 pontos com fonte de alimentação Micro800

Slots de expansão de E/S
(Aplicável somente ao Micro850)
Largura única (1º slot)
Largura dupla (2º slot)
2085-ECR (terminador)

As medições estão em mm (pol.)

Controladores Micro830 e Micro850 de 48 pontos (Lateral)



Controlador Micro830/Micro850 de 48 pontos com fonte de alimentação Micro800

As medições estão em mm (pol.)

Slots de expansão de E/S

(Aplicável somente ao Micro850)

Largura única (1º slot)

Largura dupla (2º slot)

2085-ECR (terminador)

Observações:

Fiação do controlador

Este capítulo fornece informações sobre as especificações de fiação dos controladores Micro830, Micro850 e Micro870. Inclui as seguintes seções:

Tópico	Página
Especificação da fiação e recomendações	47
Uso de supressores de transiente	48
Supressores de transientes recomendados	49
Aterrando o controlador	50
Esquemas elétricos	51
Fiação de E/S do controlador	55
Minimização do ruído elétrico	55
Orientações para fiação de canal analógico	55
Minimização do ruído elétrico nos canais analógicos	56
Aterrando o cabo analógico	56
Exemplos de fiação	57
Fiação para porta serial incorporada	58

Especificação da fiação e recomendações



ADVERTÊNCIA: Antes de você instalar e ligar a fiação a qualquer dispositivo, desligue a alimentação para o sistema do controlador.



ADVERTÊNCIA: Calcule a corrente máxima possível em cada cabo comum e de alimentação. Observe todos os códigos elétricos determinando a corrente máxima permitida para cada bitola de cabo. Uma corrente acima das taxas máximas pode fazer com que o cabo sofra sobreaquecimento, podendo causar danos.

Somente para os Estados Unidos: Se o controlador estiver instalado em um ambiente potencialmente perigoso, toda a fiação precisará estar em conformidade com as especificações declaradas no Código elétrico nacional 501-10 (b).

- Permita pelo menos 50 mm (2 pol.) entre réguas de bornes ou canaletas de fiação de E/S e o controlador.
- Direcione alimentação em entrada para o controlador por um percurso separado da fiação do dispositivo. Onde os percursos precisarem se cruzar, a sua interseção deverá ser perpendicular.



Não passe fiação de sinal ou comunicações e fiação da alimentação no mesmo eletroduto. Cabos com características de sinal diferentes devem ser passados por percursos diferentes.

- Separe a fiação por tipo de sinal. Agrupe cabos com características elétricas similares.

- Separe fiação de entrada da fiação de saída.
- Etiquete a fiação para todos os dispositivos no sistema. Utilize fita isolante, tubo termorretrátil, ou outros meios confiáveis para fins de identificação/etiquetagem dos tubos. Além de etiquetagem, utilize isolamento colorido para identificar a fiação de acordo com características de sinal. Por exemplo, você pode utilizar azul para a fiação CC e vermelho para a fiação CA.

Especificações para fiação

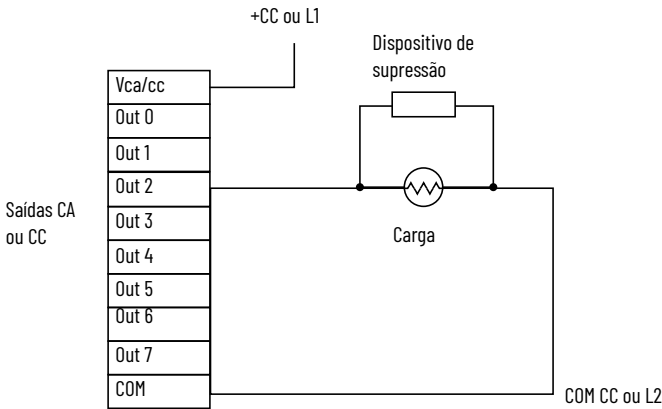
	Bitola do cabo			
	Tipo	Min.	Máx.	
Micro830Controladores Micro850/Micro870	Fio	0,2 mm ² (24 AWG)	2,5 mm ² (12 AWG)	classificado a 90 °C (194 °F) de isolamento máx
	Cabo	0,2 mm ² (24 AWG)	2,5 mm ² (12 AWG)	

Uso de supressores de transiente

Por causa dos transientes de corrente potencialmente altos que ocorrem quando se comutam dispositivos de carga indutivos, como partidas de motor e solenoides, é exigido o uso de algum tipo de supressor de transiente para proteger e prolongar a vida útil dos contatos de saída dos controladores. Acionar as cargas indutivas sem a supressão de transiente pode reduzir significativamente a vida útil dos contatos do relé. Adicionando-se um dispositivo de supressão através da bobina de um dispositivo indutivo, prolonga-se a vida da saída ou dos contatos de relé. Também são reduzidos os efeitos dos transientes de tensão e o ruído elétrico de irradiar para sistemas adjacentes.

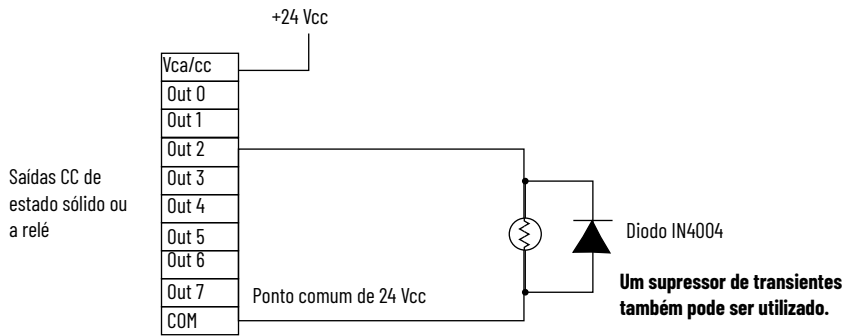
A [Figura 3](#) mostra uma saída com um dispositivo de supressão. Recomendamos que você posicione o dispositivo supressor o mais próximo possível do dispositivo de carga.

Figura 3 - Saída com um dispositivo de supressão



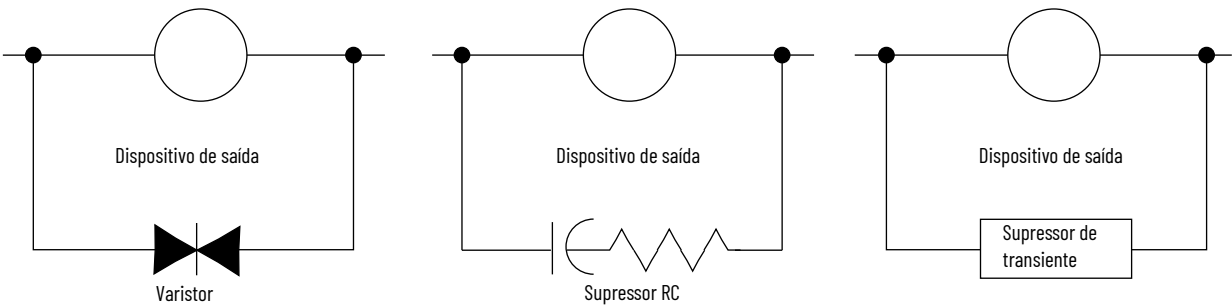
Se as saídas forem CC, recomendamos que você utilize um diodo 1N4004 para supressão de transiente, como demonstrado na [Figura 4 na página 49](#). Para dispositivos de carga CC indutiva, um diodo é adequado. Um diodo 1N4004 é aceitável para a maioria das aplicações. Um supressor de transientes também pode ser utilizado. Consulte [Supressores de transientes recomendados na página 49](#). Os circuitos de supressão de transientes são conectados diretamente através do dispositivo de carga.

Figura 4 - Saídas CC com supressão de transiente



Métodos de supressão de transientes indicados para dispositivos de carga CA indutiva incluem um varistor, um supressor RC ou um supressor de transientes Allen-Bradley, todos demonstrados na [Figura 5](#). Esses componentes precisam ser classificados apropriadamente para suprimir a característica transiente de comutação do dispositivo indutivo específico. Consulte [Supressores de transientes recomendados na página 49](#) para os supressores recomendados.

Figura 5 - Supressão de transientes para dispositivos de carga CA indutiva



Supressores de transientes recomendados

Utilize os supressores de transientes Allen-Bradley demonstrados na tabela a seguir para utilização com relés, contadores e partidas.

Supressores de transientes recomendados

Dispositivo	Tensão da bobina	Código de catálogo do supressor	Tipo ⁽⁴⁾
Cód. cat. 100/104K 700K	24 a 48 Vca	100-KFSC50	RC
	110 a 280 Vca	100-KFSC280	
	380 a 480 Vca	100-KFSC480	
	12 a 55 Vca, 12 a 77 Vcc	100-KFSV55	MOV
	56 a 136 Vca, 78 a 180 Vcc	100-KFSV136	
	137 a 277 Vca, 181 a 250 Vcc	100-KFSV277	
	12 a 250 Vcc	100-KFSD250	Diodo

Supressores de transientes recomendados (Continuação)

Dispositivo	Tensão da bobina	Código de catálogo do supressor	Tipo ⁽⁴⁾
Cód. cat. 100C, (C09 a C97)	24 a 48 Vca	100-FSC48 ⁽¹⁾	RC
	110 a 280 Vca	100-FSC280 ⁽¹⁾	
	380 a 480 Vca	100-FSC480 ⁽¹⁾	
	12 a 55 Vca, 12 a 77 Vcc	100-FSV55 ⁽¹⁾	MOV
	56 a 136 Vca, 78 a 180 Vcc	100-FSV136 ⁽¹⁾	
	137 a 277 Vca, 181 a 250 Vcc	100-FSV277 ⁽¹⁾	
	278 a 575 Vca	100-FSV575 ⁽¹⁾	
	12 a 250 Vcc	100-FSV250 ⁽¹⁾	Diodo
Partida de motor cód. cat. 509 tamanho 0 a 5	12 a 120 Vca	599-K04	MOV
	240 a 264 Vca	599-KA04	
Partida de motor cód. cat. 509 tamanho 6	12 a 120 Vca	199-FSMA1 ⁽²⁾	RC
	12 a 120 Vca	199-GSMA1 ⁽³⁾	MOV
Relé R/RM cód. cat. 700	Bobina CA	Não necessário	
	24 a 48 Vcc	199-FSMA9	MOV
	50 a 120 Vcc	199-FSMA10	
	130 a 250 Vcc	199-FSMA11	
Relé tipo N, P, PK ou PH cód. cat. 700	6 a 150 Vca/cc	700-N24	RC
	24 a 48 Vca/cc	199-FSMA9	MOV
	50 a 120 Vca/cc	199-FSMA10	
	130 a 250 Vca/cc	199-FSMA11	
	6 a 300 Vcc	199-FSMZ-1	Diodo
Dispositivos eletromagnéticos diversos limitados a 35 VA selados	6 a 150 Vca/cc	700-N24	RC

(1) Códigos de catálogo para terminais sem parafuso incluem a sequência 'CR' após '100-'. Por exemplo: Cód. cat. 100-FSC48 se torna Cód. cat. 100-CRFS48; Cód. cat. 100-FSV55 se torna 100-CRFSV55; e assim por diante.

(2) Para utilização no relé de interposição.

(3) Para utilização no contator ou partida.

(4) O tipo RC não deve ser utilizado com saídas Triac. O varistor não é recomendado para utilização com saídas a relé.

Aterrando o controlador


Esse produto é destinado para instalação em uma superfície de montagem bem aterrada como um painel de metal. Consulte Industrial Automation Wiring and Grounding Guidelines, publicação [1770-4.1](#), para obter mais informações.



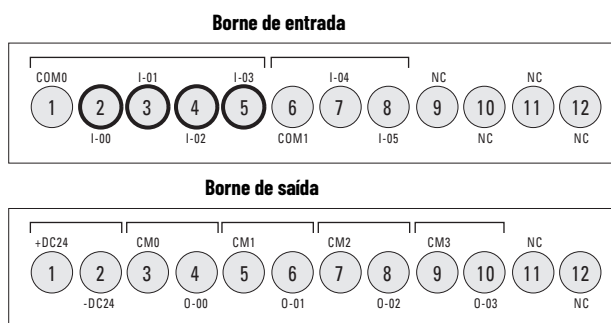
ADVERTÊNCIA: Todos os dispositivos conectados à porta de comunicação RS-232/RS-485 precisam ser referenciados ao terra do controlador, ou ficar flutuantes (não referenciados a nenhum potencial que não seja o terra). Falha ao seguir esse procedimento pode resultar em prejuízos a propriedade ou ferimentos pessoais.

Esquemas elétricos

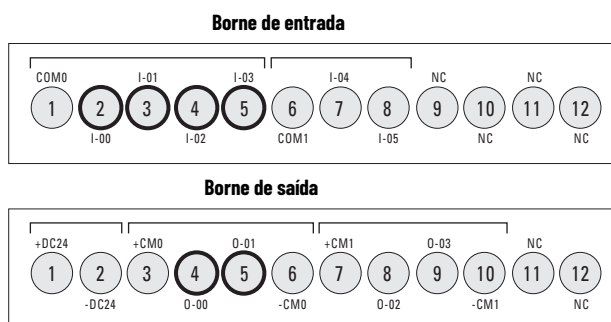
As ilustrações a seguir mostram os esquemas elétricos para os controladores Micro800. Controladores com entradas CC podem ser ligados por cabo como entradas sinking ou sourcing. Sinking e sourcing não se aplicam a entradas CA.

Entradas e saídas de alta velocidade são indicadas por .

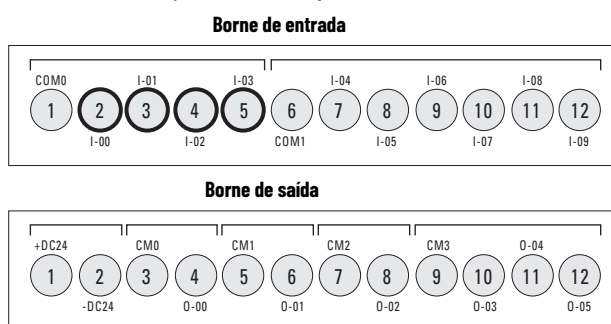
2080-LC30-10QWB



2080-LC30-10QVB

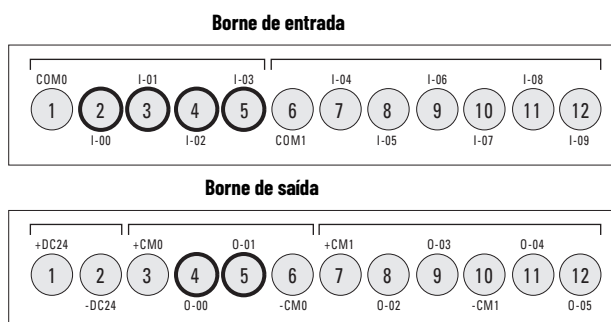


2080-LC30-16AWB, 2080-LC30-16QWB

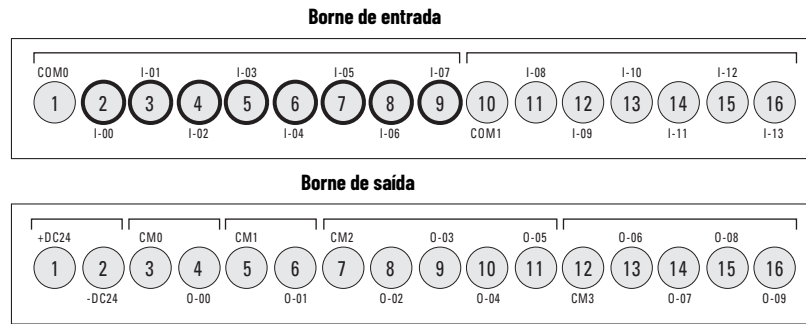


2080-LC30-16AWB não possui entradas de alta velocidade.

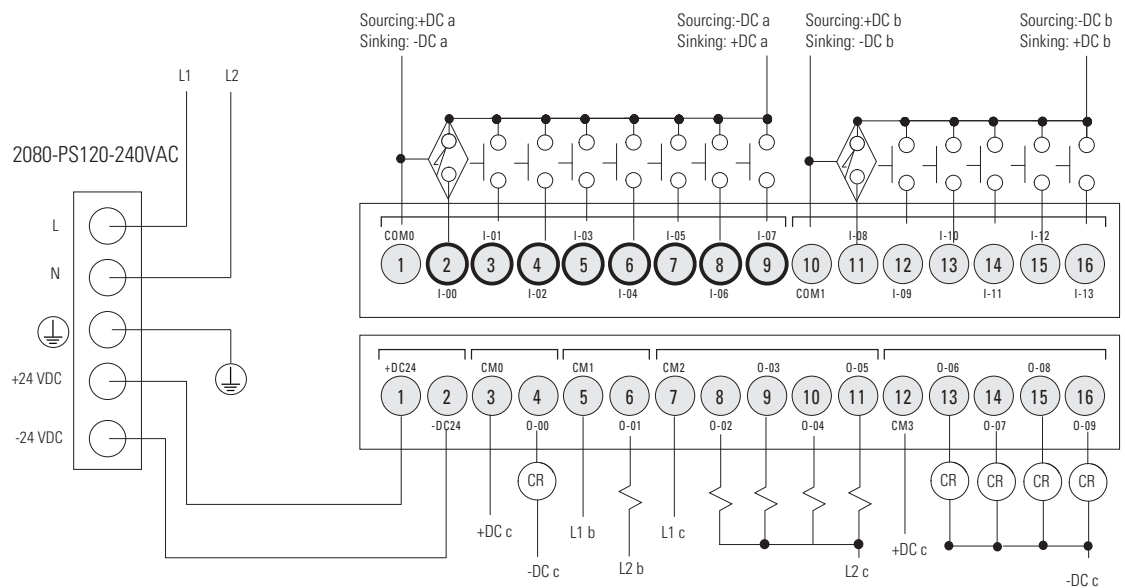
2080-LC30-16QVB



2080-LC30-24QWB, 2080-LC50-24AWB, 2080-LC50-24QWB, 2080-L50E-24AWB, 2080-L50E-24QWB, 2080-LC70-24AWB, 2080-LC70-24QWB, 2080-LC70-24QWBK, 2080-L70E-24AWB, 2080-L70E-24QWB, 2080-L70E-24QWBK, 2080-L70E-24QWBN



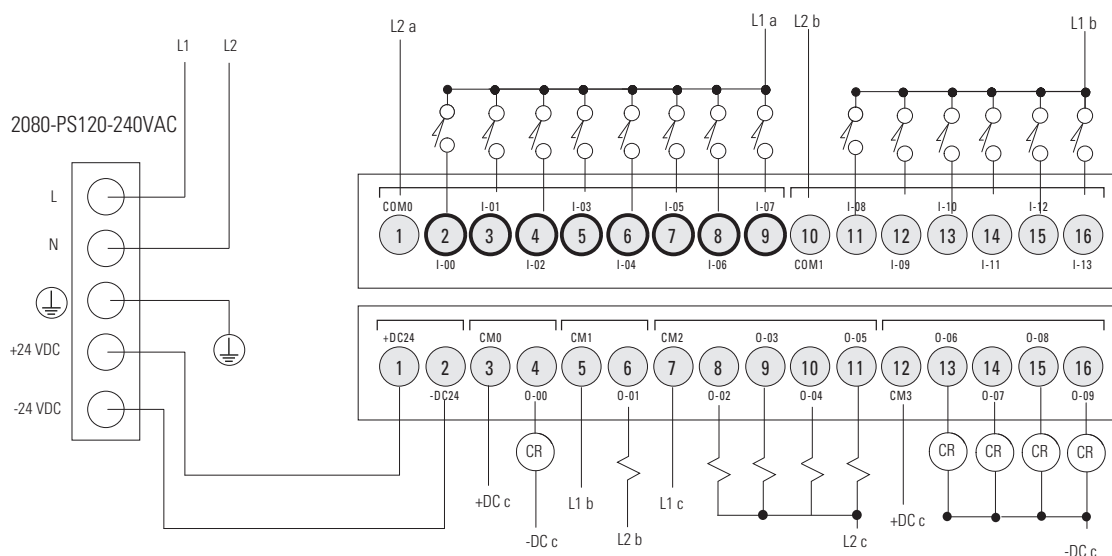
2080-LC30-24QWB, 2080-LC50-24QWB, 2080-L50E-24QWB, 2080-LC70-24QWB, 2080-LC70-24QWBK, 2080-L70E-24QWB, 2080-L70E-24QWBK, 2080-L70E-24QWBN, Configuração de entrada CC



IMPORTANTE

- Não conecte -DC24 (terminal de saída 2) ao terra/chassi aterrado.
- Nos sistemas Micro870 que usam mais de quatro módulos de expansão de E/S Micro800, recomendamos usar uma fonte de alimentação 1606-XLP60EQ em vez de uma fonte de alimentação 2080-PS120-240VAC. Conecte o controlador Micro870 e a fonte de alimentação de expansão 2085-EP24VDC à mesma fonte de alimentação 1606-XLP60EQ.

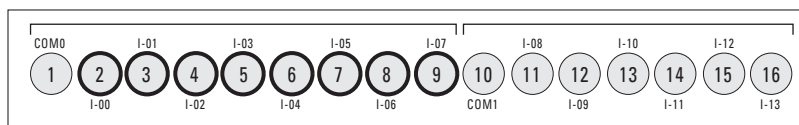
**2080-LC50-24AWB, 2080-L50E-24AWB, 2080-LC70-24AWB, 2080-L70E-24AWB,
Configuração da entrada CC**



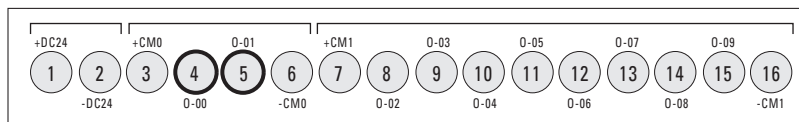
IMPORTANTE Não conecte -DC24 (terminal de saída 2) ao terra/chassi aterrado.

**2080-LC30-24QVB, 2080-LC30-24QBB, 2080-LC50-24QVB, 2080-LC50-24QBB, 2080-L50E-24QVB,
2080-L50E-24QBB, 2080-LC70-24QBB, 2080-LC70-24QBBK, 2080-L70E-24QBB, 2080-L70E-24QBBK,
2080-L70E-24QBBN**

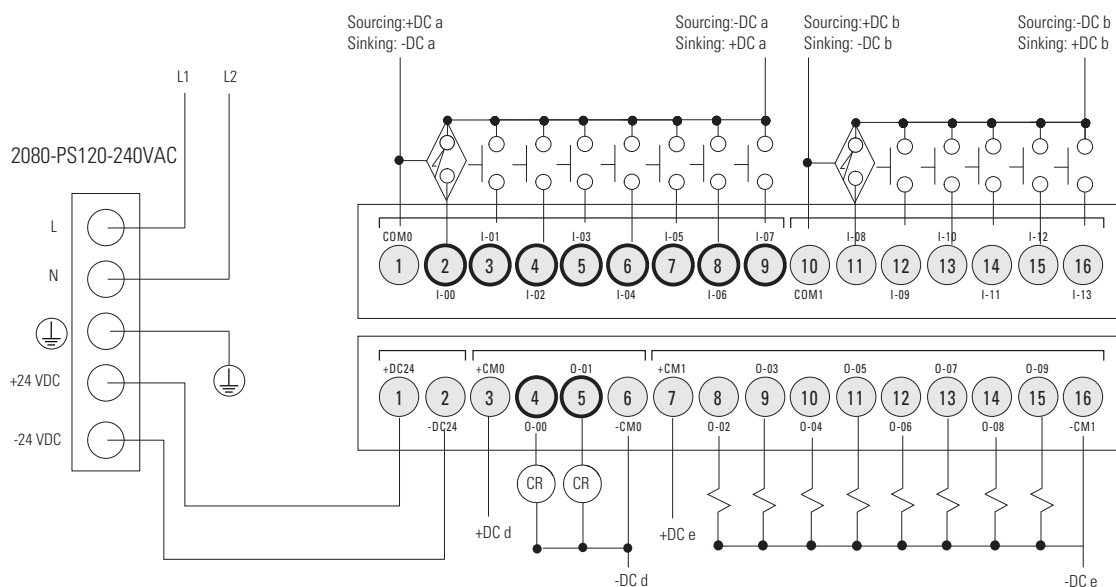
Borne de entrada



Borne de saída



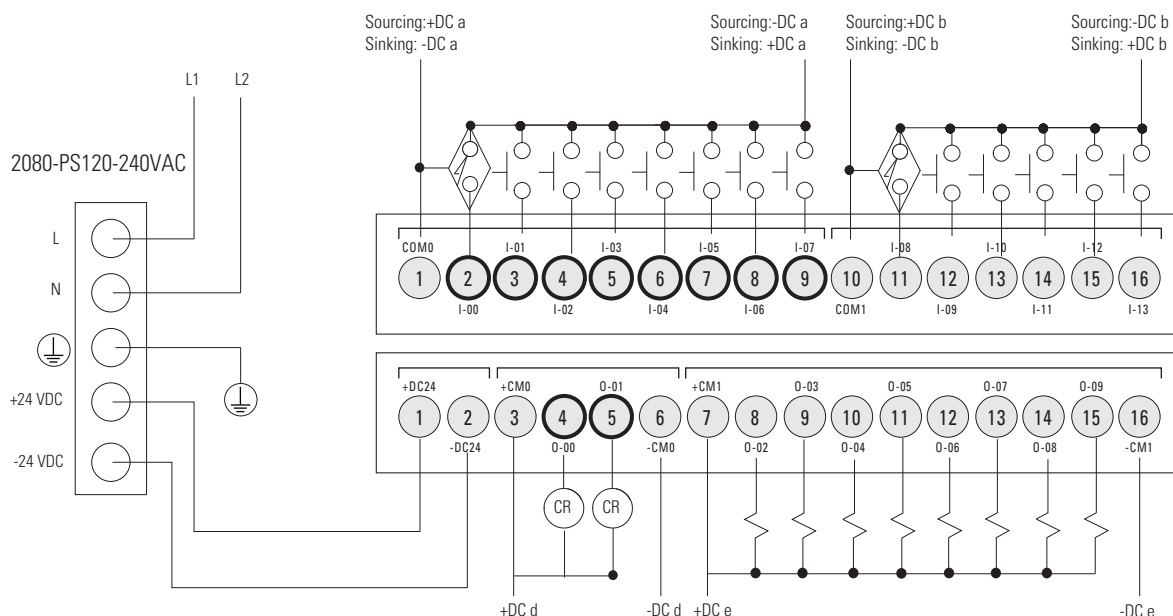
**2080-LC30-24QBB, 2080-LC50-24QBB, 2080-L50E-24QBB, 2080-LC70-24QBB, 2080-LC70-24QBBK,
2080-L70E-24QBB, 2080-L70E-24QBBK, 2080-L70E-24QBBN,
Configuração da entrada CC**



IMPORTANTE

- Não conecte -DC24 (terminal de saída 2) ao terra/chassi aterrado.
- Nos sistemas Micro870 que usam mais de quatro módulos de expansão de E/S Micro800, recomendamos usar uma fonte de alimentação 1606-XLP60EQ em vez de uma fonte de alimentação 2080-PS120-240VAC. Conecte o controlador Micro870 e a fonte de alimentação de expansão 2085-EP24VDC à mesma fonte de alimentação 1606-XLP60EQ.

2080-LC30-24QVB, 2080-LC50-24QVB, 2080-L50E-24QVB, Configuração de entrada CC

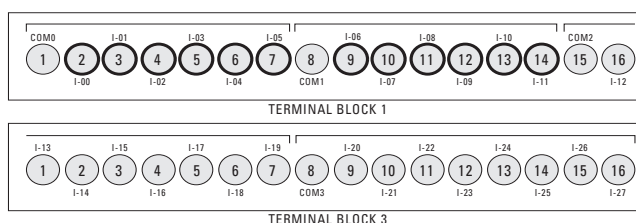


IMPORTANTE

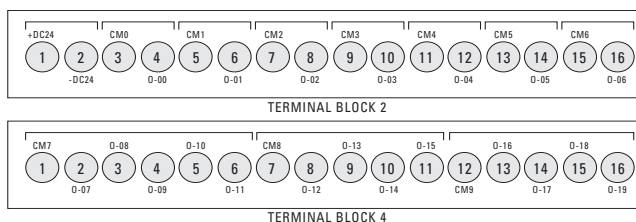
Não conecte -DC24 (terminal de saída 2) ao terra/chassi aterrado.

2080-LC30-48AWB, 2080-LC30-48QWB, 2080-LC50-48AWB, 2080-LC50-48QWB, 2080-LC50-48QWBK, 2080-L50E-48AWB, 2080-L50E-48QWB, 2080-L50E-48QWBK

Bornes de entrada



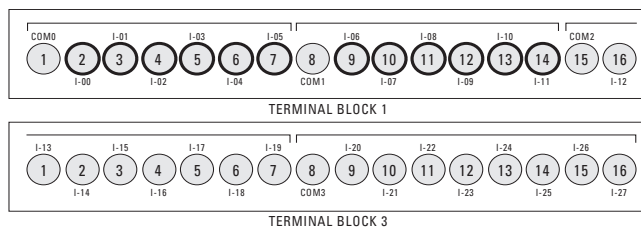
Bornes de saída



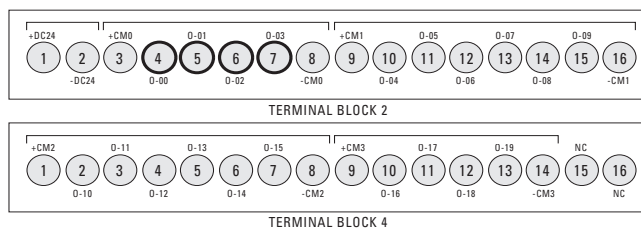
2080-LC30-48AWB não possui entradas de alta velocidade.

2080-LC30-48QVB, 2080-LC30-48QBB, 2080-LC50-48QVB, 2080-LC50-48QBB, 2080-L50E-48QVB, 2080-L50E-48QBB

Bornes de entrada



Bornes de saída



Fiação de E/S do controlador

Esta seção contém algumas informações relevantes sobre como minimizar ruído elétrico e também inclui alguns exemplos de fiação.

Minimização do ruído elétrico

Por causa da variedade de aplicações e ambientes em que os controladores são instalados e ficam operando, é impossível garantir que todo o ruído ambiental será removido pelos filtros de entrada. Para auxiliar a reduzir os efeitos do ruído ambiental, instale o sistema Micro800 em um gabinete apropriadamente classificado (por exemplo, NEMA). Verifique se o sistema Micro800 está adequadamente aterrado.

Um sistema pode apresentar falhas devido a uma alteração no ambiente de operação após um período de tempo. Recomendamos verificar periodicamente a operação do sistema, especialmente quando novas máquinas ou outras fontes de ruído forem instaladas próximas ao sistema Micro800.

Orientações para fiação de canal analógico

Considere o seguinte quando for realizar a fiação de seus canais analógicos:

- O ponto comum analógico (COM) não é isolado eletricamente do sistema e é conectado ao ponto comum da fonte de alimentação.
- Canais analógicos não são isolados uns dos outros.
- Utilize cabo blindado Belden #8761 ou equivalente.
- Sob condições normais, o cabo dreno (blindagem) deve ser conectado ao painel metálico de montagem (ligação à terra). Mantenha a conexão de blindagem à terra o mais curta possível.
- Para garantir precisão ideal para entradas de tipo de tensão, limite a impedância geral do cabo mantendo todos os cabos analógicos o mais curtos possível. Posicione o sistema E/S o mais próximo possível dos seus sensores de tipo de tensão ou atuadores.

Minimização do ruído elétrico nos canais analógicos

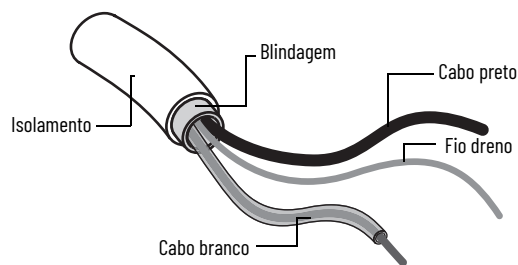
Entradas em canais analógicos utilizam filtros de alta frequência digitais que reduzem significativamente os efeitos do ruído elétrico nos sinais de entrada. Porém, devido à variedade de aplicações e ambientes em que os controladores analógicos são instalados e são operados, é impossível garantir que todo o ruído ambiental será removido pelos filtros de entrada.

Diversos passos específicos podem ser tomados para auxiliar a redução dos efeitos do ruído ambiental nos sinais analógicos:

- Instale o sistema Micro800 em um gabinete apropriadamente classificado, por exemplo, NEMA. Certifique-se de que a blindagem esteja adequadamente aterrada.
- Utilize cabo Belden #8761 para ligar os canais analógicos, certificando-se de que o cabo dreno e a blindagem estejam devidamente aterrados.
- Passe o cabo Belden por um percurso separado de qualquer fiação CA. Imunidade a ruído adicional pode ser obtida passando os cabos por um eletroduto aterrado.

Aterrando o cabo analógico

Utilize cabo de comunicação blindado (Belden #8761). O cabo Belden possui dois cabos de sinal (preto e branco), um cabo dreno, e uma blindagem. O cabo dreno e a blindagem devem ser aterrados em uma extremidade do cabo.

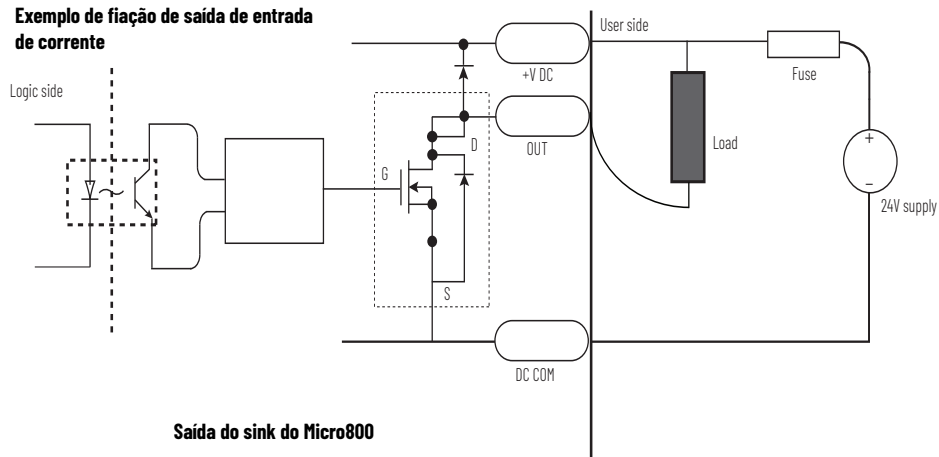


IMPORTANTE Não realize o aterramento do cabo dreno e da blindagem em ambas as terminações do cabo.

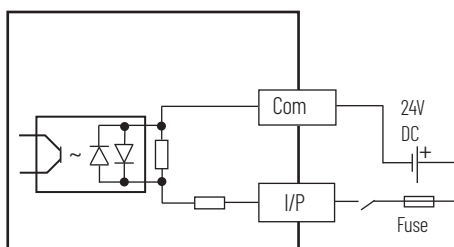
Exemplos de fiação

Exemplos de fiações de sink/source, entrada/saída são demonstrados abaixo.

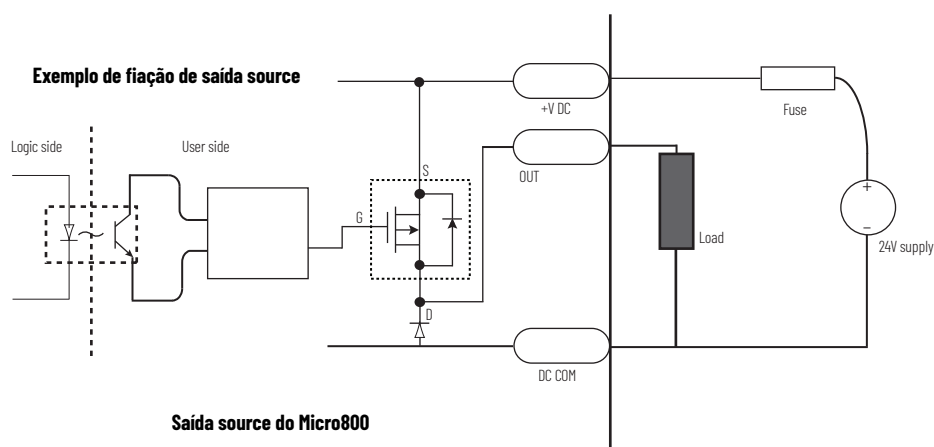
Exemplo de fiação de saída de entrada de corrente



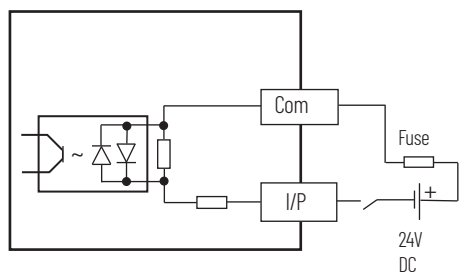
Exemplo de fiação de entrada sink



Exemplo de fiação de saída source



Exemplo de fiação de entrada source



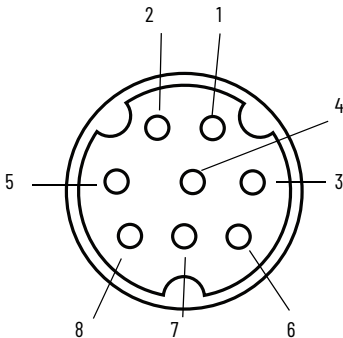
Fiação para porta serial incorporada

A porta serial incorporada é de tipo RS-232/RS-485, não isolada, destinada à utilização em distâncias curtas (<3 m) a dispositivos como IHMs.

Consulte [Cabos para porta serial incorporada na página 22](#) para uma lista de cabos que podem ser utilizados com o conector mini DIN de 8 pinos da porta serial incorporada.

Por exemplo, o cabo 1761-CBL-PM02 é utilizado geralmente para conectar a porta serial incorporada ao PanelView™ 800 IHM utilizando a RS-232.

Porta serial incorporada



Explicações sobre pinagem

Pino	Definição	Exemplo com RS-485	Exemplo com RS-232
1	RS-485+	B+	(não utilizado)
2	TERRA	TERRA	TERRA
3	RS-232 RTS	(não utilizado)	RTS
4	RS-232 RxD	(não utilizado)	RxD
5	RS-232 DCD	(não utilizado)	DCD
6	RS-232 CTS	(não utilizado)	CTS
7	RS-232 TxD	(não utilizado)	TxD
8	RS-485-	A(-)	(não utilizado)

IMPORTANTE

- Não conecte o pino GND da porta serial ao terra/chassi aterrado. O pino GND da porta serial é a CC comum para os sinais de comunicação da porta serial e não se destina ao aterramento da blindagem.
- Se o comprimento do cabo serial for maior que 3 metros, use uma porta serial isolada, código de catálogo 2080-SERIALISOL.

Conexões de comunicação

Visão geral

Este capítulo descreve como comunicar-se com o seu sistema de controle e configurar as definições de comunicação. O método que você utiliza e a fiação necessária para conectar o seu controlador depende do tipo de sistema que você está empregando. Este capítulo também descreve como o controlador estabelece comunicação com a rede apropriada. Os temas incluem:

Tópico	Página
Protocolos de comunicação suportados	59
Transmissão de comunicação CIP	64
Usar modems com controladores Micro800	65
Configuração da porta serial	66
Configurar as definições de Ethernet	72
Configurar o driver serial CIP	74
Suporte OPC usando FactoryTalk Linx	75

Os controladores Micro830, Micro850 e Micro870 possuem os seguintes canais de comunicação incorporados:

- uma porta combo RS-232/RS-485 não isolada
- uma porta de programação USB não isolada

Além disso, os controladores Micro850 e Micro870 têm uma porta Ethernet RJ-45.

Protocolos de comunicação suportados

Os controladores Micro830/Micro850 e Micro870 suportam protocolos de comunicação por meio da porta serial incorporada RS-232/RS-485, bem como quaisquer módulos de encaixe de porta serial instalados. Além disso, os controladores Micro850 também suportam comunicação por porta Ethernet incorporada, e podem ser conectados a uma rede local para diversos dispositivos fornecendo taxa de transferência de 10 Mbps/100 Mbps.

Estes são protocolos de comunicação suportados por controladores Micro830/Micro850/Micro870:

- Modbus RTU mestre e escravo
- Cliente/servidor CIP serial (DF1)
- Cliente/servidor CIP Symbolic
- ASCII
- DNP3 ⁽¹⁾

(1) DNP3 é suportado somente em controladores 2080-L70E-24QxBN.

Estes são protocolos de comunicação suportados somente por controladores Micro850 e Micro870:

- EtherNet/IP Cliente/Servidor
- Cliente/Servidor Modbus/TCP
- Cliente DHCP
- Soquetes cliente/servidor TCP/UDP

Limites de conexão para controladores Micro830/Micro850/Micro870

Descrição		Micro830	Micro850/ Micro870
Conexões CIP			
Número total de conexões de clientes mais servidores para todas as portas		16	24
Número máximo de conexões de cliente para todas as portas		15	16
Número máximo de conexões de servidor para todas as portas		16	24
Número máximo de conexões EtherNet/IP	Cliente	-	16
	Servidor		23
Número máximo de conexões USB	Cliente	-	-
	Servidor	15	23
Número máximo de conexões seriais	Cliente	15	16
	Servidor	15	23
Conexões TCP			
Número total de conexões de clientes mais servidores		-	64
Número máximo para EtherNet/IP	Cliente		16
	Servidor		16
Número máximo para Modbus TCP	Cliente		16
	Servidor		16
Número máximo para soquetes programáveis de usuário			8
Soquetes programáveis de usuário			
Número total de soquetes programáveis de usuário(qualquer combinação de UDP mais TCP Cliente/servidor)		-	8

IMPORTANTE Se todas as conexões cliente/servidor estiverem totalmente carregadas, o desempenho pode ser afetado, com perda de dados e atrasos intermitentes durante a comunicação.

Eis alguns exemplos de configuração com base nos limites descritos na tabela acima:

1. O número máximo de inversores que podem ser controlados por EtherNet/IP é 16. Isso se deve ao limite máximo de conexões TCP cliente, que é 16, e ao limite máximo de conexões EtherNet/IP Cliente, que também é 16.
2. Se você tiver 10 dispositivos controlados por EtherNet/IP, o número máximo de dispositivos que podem ser controlador por serial será seis. Isso ocorre porque o limite máximo de conexões de cliente é 16.
3. O número total de soquetes UDP mais soquetes TCP Cliente/Servidor tem um limite máximo de oito.

Modbus RTU

Modbus é um protocolo de comunicação halfduplex mestre-escravo. O mestre da rede Modbus lê e grava bits e registros. O protocolo Modbus permite que um único mestre se comunique com um máximo de 247 dispositivos escravos. Controladores Micro800 suportam protocolos Modbus RTU mestre e escravo. Para mais informações sobre a configuração do seu controlador Micro800 para protocolo Modbus, consulte a ajuda on-line do software Connected Components Workbench. Para mais informações sobre o protocolo Modbus, consulte as especificações do protocolo Modbus disponíveis em <https://www.modbus.org>.

Para obter informações sobre o mapeamento Modbus, consulte [Mapeamento Modbus para Micro800 na página 259](#).

Para configurar a porta serial como Modbus RTU, consulte [Configuração Modbus RTU na página 70](#).



Use a instrução MSG_MODBUS para enviar mensagens Modbus através da porta serial.

Cliente/servidor CIP serial – DF1

O Cliente/Servidor CIP serial permite que o protocolo CIP seja usado por meio de uma porta serial. Ele é usado tipicamente com modems. A vantagem sobre protocolos seriais não CIP é que, como o protocolo é CIP, são suportados downloads de programas, inclusive CIP pass-through da porta serial para Ethernet.

ASCII

ASCII fornece conexão a outros dispositivos ASCII, como leitores de código de barras, balanças, impressoras seriais e outros dispositivos inteligentes. Você pode usar o ASCII configurando a porta serial RS-232/R-S485 plug-in ou incorporada para o driver ASCII. Consulte a ajuda on-line do software Connected Components Workbench para mais informações.

Para configurar a porta serial para ASCII, consulte [Configurar ASCII na página 71](#).

Cliente/Servidor Modbus/TCP

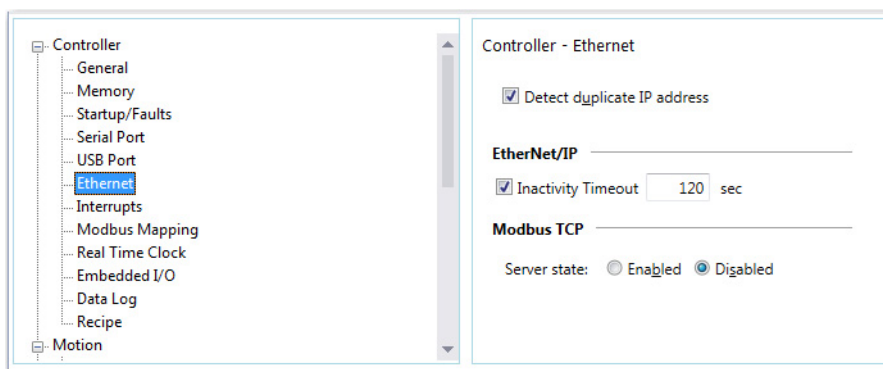
O protocolo de comunicação cliente/servidor Modbus TCP usa os mesmos recursos de mapeamento que o Modbus RTU, mas em vez de porta serial, é suportado por Ethernet. O servidor Modbus/TCP incorpora os recursos do Modbus escravo na Ethernet.

Nenhuma configuração de protocolo é requerida além de configurar a tabela de mapeamento Modbus. Para obter informações sobre o mapeamento Modbus, consulte [Mapeamento Modbus para Micro800 na página 259](#).



Use a instrução MSG_MODBUS2 para enviar mensagens Modbus TCP pela porta Ethernet.

Com o software Connected Components Workbench versão 12 ou posterior, o servidor Modbus TCP é desativado por padrão. Se você quiser usar o Modbus TCP, você pode ativá-lo nas configurações Ethernet.



Cliente/servidor CIP Symbolic

O CIP Symbolic é suportado por qualquer interface compatível com CIP incluindo Ethernet (EtherNet/IP) e porta serial (CIP Serial). Esse protocolo permite que IHMs se conectem facilmente ao controlador Micro830/Micro850/Micro870.

Os controladores Micro850 e Micro870 suportam até 16 conexões simultâneas do cliente EtherNet/IP e 23 conexões simultâneas do servidor EtherNet/IP.

O CIP serial, suportado nos controladores Micro830, Micro850 e Micro870, faz uso de protocolo DF1 full-duplex, que oferece conexão ponto a ponto entre dois dispositivos.

DF1 half-duplex mestre, DF1 half-duplex escravo e Modem de rádio DF1 são suportados pelos controladores Micro850 (2080-L50E) e Micro870 (2080-L70E). Esses protocolos fornecem conexão a vários dispositivos por meio de RS-485 ou modems de rádio.

Os controladores Micro800 suportam o protocolo por conexão RS-232 a dispositivos externos, como computadores executando software RSLinx® Classic, terminais PanelView Component (versão do firmware 1.70 ou superior), terminais PanelView 800 ou outros controladores que suportem CIP Serial por DF1 full-duplex, como controladores ControlLogix e CompactLogix que tenham portas seriais incorporadas. Os cód. cat. 2080-L50E e 2080-L70E também suportam protocolo DF1 half-duplex e Modem de Rádio.

A EtherNet/IP, suportada pelo controlador Micro850 e Micro870, usa o protocolo Ethernet TCP/IP padrão.

Os controladores Micro850 e Micro870 suportam até 23 conexões de servidor EtherNet/IP simultâneas.

Para configurar o CIP Serial, consulte [Configurar o driver serial CIP na página 66](#).

Para configurar para EtherNet/IP, consulte [Configurar as definições de Ethernet na página 72](#).

Para obter mais informações sobre o protocolo DF1, consulte [Conexão a redes usando DF1 na página 335](#).

Endereçamento CIP Symbolic

Usuários podem acessar quaisquer variáveis globais por meio de endereçamento CIP Symbolic exceto por variáveis de sistema e reservadas.

São suportados vetores de uma ou duas dimensões para tipos de dados simples (por exemplo, ARRAY OF INT[1...10, 1...10]), mas não são suportados vetores de vetores (por exemplo, ARRAY OF ARRAY). Vetores de grupos também são compatíveis.

Tipos de dados suportados em CIP Symbolic

Tipo de dados ⁽¹⁾	Descrição
BOOL	Lógica booleana com valores VERDADEIRO (1) e FALSO (0) (Usa até 8 bits de memória)
SINT	Inteiro com sinal e de 8 bits
INT	Inteiro com sinal e de 16 bits
DINT	Inteiro com sinal e de 32 bits
LINT ⁽²⁾	Inteiro com sinal e de 64 bits
USINT	Inteiro sem sinal e de 8 bits
UINT	Inteiro sem sinal e de 16 bits
UDINT	Inteiro sem sinal e de 32 bits
ULINT ⁽²⁾	Inteiro sem sinal e de 64 bits
REAL	Valor de ponto flutuante de 32 bits
LREAL ⁽²⁾	Valor de ponto flutuante de 64 bits
STRING	Grupo de caracteres (1 byte por caractere)
DATA ⁽³⁾	Inteiro sem sinal e de 32 bits
HORA ⁽³⁾	Inteiro sem sinal e de 32 bits

(1) Instrução MSG Logix pode ler/gravar os tipos de dados SINT, INT, DINT, LINT e REAL utilizando tipos de mensagem de "leitura de tabela de dados CIP" e "gravação de tabela de dados CIP". Os tipos de dados BOOL, USINT, UINT, UDINT, ULINT, LREAL, STRING, SHORT_STRING, DATE e TIME não podem ser acessados com instruções Logix MSG.

(2) Não compatível com PanelView Component ou PanelView 800.

(3) Pode ser usado enviando dados para UDINT, principalmente para uso com terminais IHM PanelView Plus e PanelView 800.

Mensagem de cliente CIP

As mensagens CIP Generic e CIP Symbolic são suportadas pelos controladores Micro800 pela Ethernet e as portas seriais. Esses recursos de envio de mensagens do cliente são habilitados pelos blocos de funções MSG_CIPSYMBOLIC e MSG_CIPGENERIC.

Para mais informações e exemplos de projeto de início rápido para auxiliá-lo a usar o recurso Envio CIP Client Messaging, consulte Controladores programáveis Micro800: Iniciando com a mensagem de cliente CIP, publicação [2080-QS002](#).

Soquetes cliente/servidor TCP/UDP

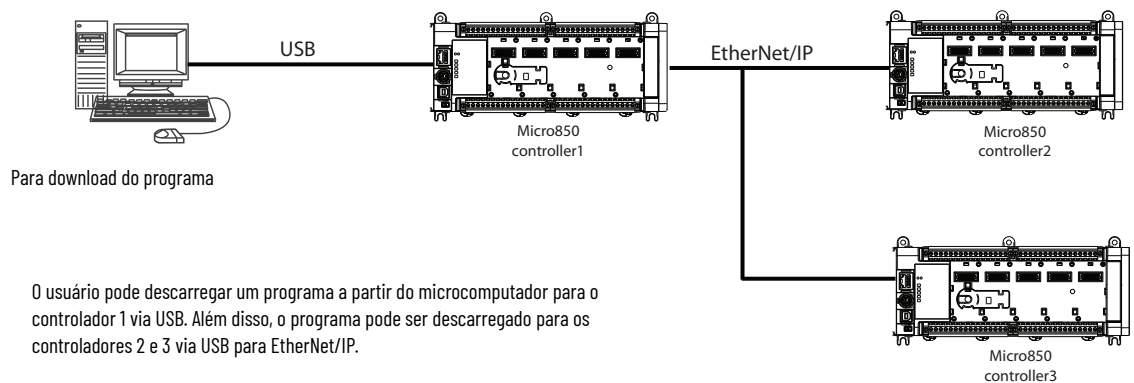
O protocolo de soquetes é usado para comunicações Ethernet para dispositivos que não suportam Modbus TCP e EtherNet/IP. Os soquetes suportam cliente e servidor, além de TCP e UDP. As aplicações típicas incluem comunicação com impressoras, leitores de código de barras e PCs.

Transmissão de comunicação CIP

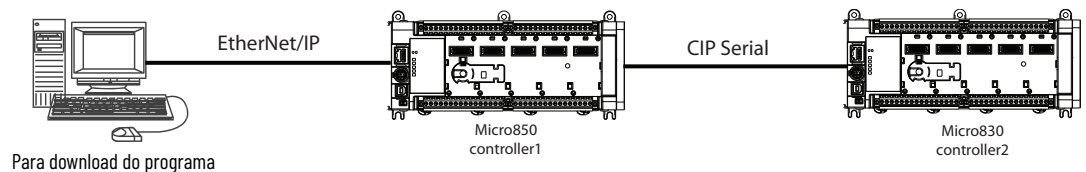
Os controladores Micro830, Micro850 e Micro870 suportam transmissão de comunicação em qualquer porta de comunicação que suporte Common Industrial Protocol (CIP) para aplicações como download de programas. Eles não suportam aplicações que exigem conexões dedicadas como IHM. O Micro830, o Micro850 e o Micro870 suportam no máximo um salto entre redes. Um salto entre redes é definido como sendo uma conexão intermediária ou link de comunicação entre dois dispositivos – no Micro800, isso é por meio de EtherNet/IP, CIP Serial ou USB CIP.

Exemplos de arquiteturas suportadas

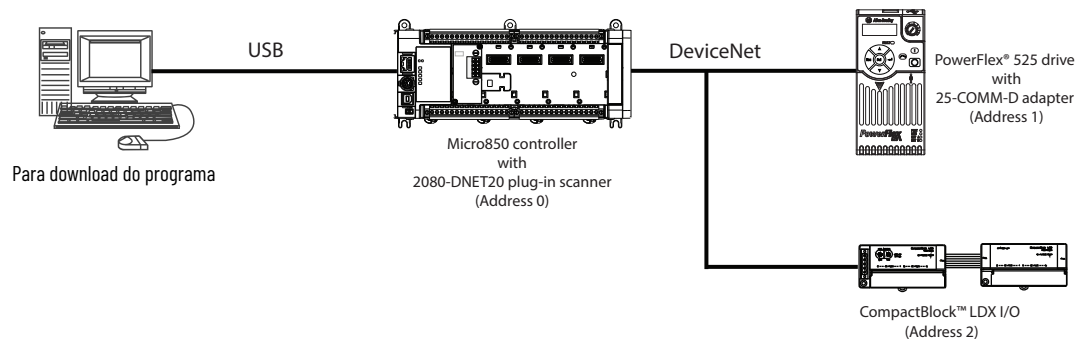
USB para EtherNet/IP



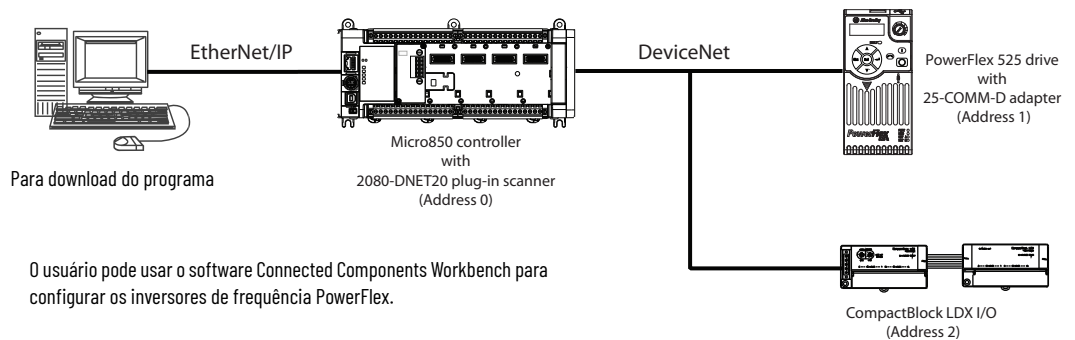
EtherNet/IP para serial CIP



USB para DeviceNet



EtherNet/IP para DeviceNet



IMPORTANTE Os controladores Micro800 não suportam mais de um salto entre redes (por exemplo, da Ethernet/IP -> CIP Serial -> Ethernet/IP).

Usar modems com controladores Micro800

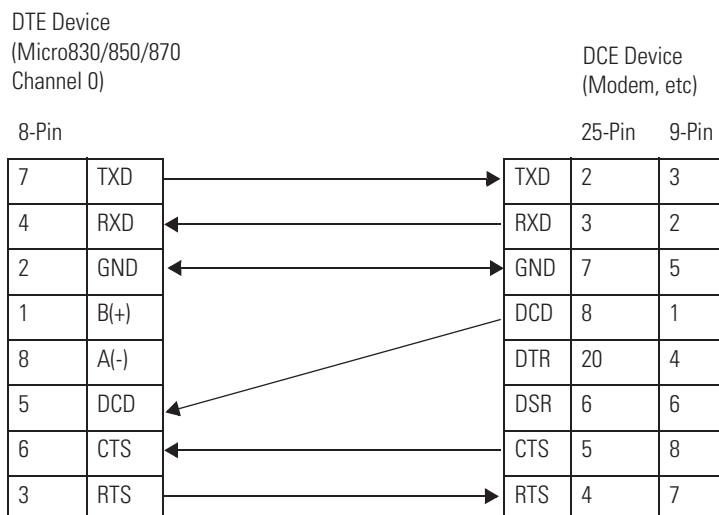
Modems seriais podem ser utilizados com os controladores Micro830, Micro850 e Micro870.

Realizando uma conexão DF1 ponto a ponto

Você pode conectar os controladores programáveis Micro830, Micro850 e Micro870 ao seu modem serial utilizando um cabo serial para modem nulo Allen-Bradley (1761-CBL-PM02) na porta serial incorporada do controlador junto com um adaptador para modem nulo de 9 pinos - um modem nulo com um adaptador para modem nulo é equivalente a um cabo para modem. O protocolo recomendado para essa configuração é o CIP Serial.

Fabrique o seu próprio cabo para modem

Se você fabricar o seu próprio cabo para modem, o comprimento máximo do cabo deve ser de 15,24 m (50 pés) com um conector de 25 ou 9 pinos. Consulte a pinagem típica a seguir para fabricar um cabo direto:

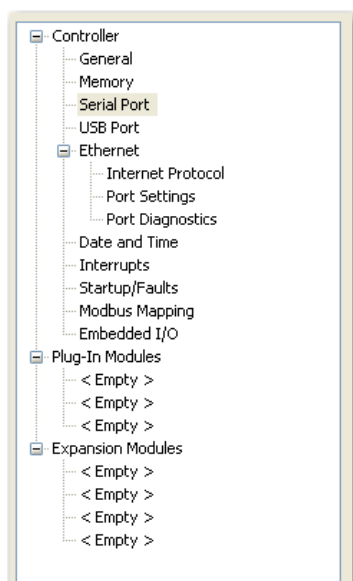


Configuração da porta serial

Você pode configurar o driver da porta serial como CIP Serial, Modbus RTU, ASCII ou Shutdown por meio da árvore de configuração do dispositivo no software Connected Components Workbench.

Configurar o driver serial CIP


1. Abra o seu projeto software Connected Components Workbench. Na árvore de configuração do dispositivo, vá até Controller properties. Clique em Serial Port.



2. Selecione CIP Serial a partir do campo Driver.

Controller - Serial Port

Common Settings

Driver: CIP Serial 

Baud Rate: 38400

Parity: None

Station Address: 1

Protocol Control

DF1 Mode: Full Duplex

Media: RS232

Control Line: No Handshake

Error Detection: CRC

Embedded Responses: After One Received

☒ Duplicate Packet Detection

ACK Timeout: 50 ×20 ms ENQ Retries: 3

NAK Retries: 3

3. Especifique uma baud rate. Selecione uma taxa de comunicação que seja suportada por todos os dispositivos no seu sistema. Configure todos os dispositivos no sistema para a mesma taxa de comunicação. A velocidade de transmissão padrão é definida como 38.400bps.
4. Na maioria dos casos, os parâmetros parity e station address devem ser deixados nas suas configurações padrão.
5. Clique em Advanced Settings e defina os parâmetros avançados. Consulte a [Tabela 5](#) para uma descrição dos parâmetros CIP Serial.

Tabela 5 - Parâmetros do driver serial CIP

Parâmetro	Opções	Padrão
Baud Rate	Alterna entre as velocidades de comunicação de 1200, 2400, 4800, 9600, 19200 e 38400.	38400
Paridade	Especifica a configuração de paridade para a porta serial. Paridade fornece detecção adicional de erro em mensagem-pacote. Selecione Even, Odd, ou None.	Nenhuma
Endereço da estação (Nó)	Insira um valor entre 0 e 254. Insira 1 para DF1 full-duplex.	1
Modo DF1 ⁽¹⁾	Define o modo DF1 - full-duplex, half-duplex mestre, half-duplex escravo, Modem de rádio ⁽²⁾ .	Configurado como full-duplex por padrão.
Control Line	<ul style="list-style-type: none"> Full-duplex: sem reconhecimento, full-duplex (RTS sempre ligado). Half-duplex escravo: sem reconhecimento, half-duplex sem portadora contínua (RTS/CTS). Half-duplex mestre: sem reconhecimento, half-duplex sem portadora contínua (RTS/CTS), full-duplex (RTS sempre ligado). Modem de rádio: sem reconhecimento, half-duplex sem portadora contínua (RTS/CTS), half-duplex com reconhecimento DCD. 	Configurado como sem handshake por padrão.
Duplicar detecção de pacote	Detecta e elimina respostas duplicadas a uma mensagem. Pacotes duplicados podem ser enviados sob condições de comunicação com ruído quando as novas tentativas do emissor não estiverem configuradas como 0. Alterna entre Enabled e Disabled.	Habilitado
Error Detection	Alterna-se entre CRC e BCC.	CRC

Tabela 5 - Parâmetros do driver serial CIP (Continuação)

Parâmetro	Opções	Padrão
Respostas incorporadas	Para utilizar respostas integradas, escolha Enabled Unconditionally. Se você deseja que o controlador utilize respostas integradas somente quando detectar respostas integradas de outro dispositivo, escolha After One Received. Se você estiver se comunicando com outro dispositivo Allen-Bradley, escolha Enabled Unconditionally. Respostas integradas aumentam a eficiência do tráfego da rede.	After One Received
Repetições NAK	O número de vezes que o controlador reenviará um pacote de mensagem porque o controlador recebeu uma resposta NAK à transmissão de pacote de mensagem anterior.	3
ENQ Retries	O número de consultas (ENQs) que você deseja que o controlador envie após uma expiração de tempo ACK.	3
Transmit Retries	Especifica o número de vezes que tenta-se reenviar uma mensagem após a primeira tentativa antes de ter sido declarada como impossível de enviar. Insira um valor entre 0 e 127.	3
Tempo-limite ACK (x20 ms)	Especifica a quantidade de tempo esperada para um ACK após um pacote ser transmitido.	50
Supressão EOT	Habilitada, Desabilitada Quando a Supressão de EOT está habilitada, o escravo não responde quando consultado se nenhuma mensagem estiver enfileirada. Isso economiza energia e tempo de transmissão do modem quando não há mensagem para transmitir.	Desabilitado
Tempo limite de coleta (x20 ms)	0 a 65.535 (pode ser definido em incrementos de 20 ms) O tempo limite de consulta se aplica somente quando um dispositivo escravo inicia uma instrução MSG. É a quantidade de tempo que o dispositivo escravo aguarda por uma pesquisa do dispositivo mestre. Se o dispositivo escravo não receber uma consulta dentro do tempo limite de consulta, é gerado um erro de instrução MSG e o programa ladder precisa colocar novamente a instrução MSG na fila. Se você estiver usando uma instrução MSG, é recomendado que não seja usado um tempo limite de consulta. O tempo limite de consulta é desabilitado quando definido como zero.	3000
Atraso na desenergização de RTS (x20 ms)	0 a 65.535 (pode ser definido em incrementos de 20 ms) Especifica o tempo de atraso entre o momento em que o último caractere serial é enviado ao modem e o momento em que o RTS é desativado. Dá ao modem tempo extra para transmitir o último caractere de um pacote.	0
Atraso de envio RTS (x20 ms)	0 a 65.535 (pode ser definido em incrementos de 20 ms) Especifica o tempo de atraso entre a configuração RTS e a checagem pela resposta CTS. Para utilização com modems que não estão prontos para responder com CTS imediatamente após receber o RTS.	0
Nova tentativa de mensagem	0 a 255 Especifica o número de vezes que um dispositivo escravo tenta reenviar um pacote de mensagem quando não recebe um ACK do dispositivo mestre. Para uso em ambientes com ruído em que os pacotes de mensagem podem ser corrompidos na transmissão.	3
Faixa de coleta prioritária - alta	Selecione o endereço da última estação escrava para a coleta prioritária.	0
Faixa de coleta prioritária - baixa	Selecione o endereço da primeira estação escrava para a coleta prioritária. Inserir 255 desabilita a coleta prioritária.	255
Faixa de coleta normal - alta	Selecione o endereço da última estação escrava para a coleta normal.	0
Faixa de coleta normal - baixo	Selecione o primeiro endereço da estação escrava para coleta normal. Inserir 255 desabilita a coleta normal.	255

Tabela 5 - Parâmetros do driver serial CIP (Continuação)

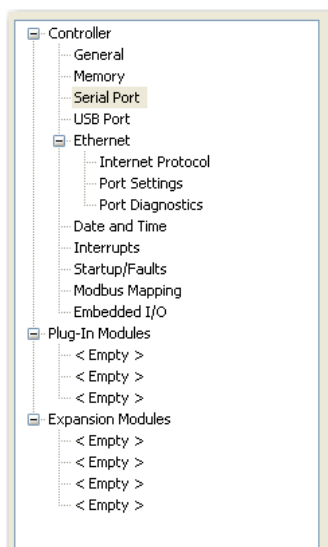
Parâmetro	Opções	Padrão
Tamanho do grupo de coleta normal	Insira a quantidade de estações ativas localizadas na faixa de coleta normal que você deseja fazer durante uma varredura na faixa de coleta de dados normal antes de retornar à faixa de coleta prioritária. Se nenhuma estação estiver configurada na Priority Polling Range, deixe este parâmetro com 0.	0
Tempo limite de espera da mensagem de resposta	Define a quantidade de tempo, em incrementos de 20 ms, que a estação mestre irá esperar após receber um ACK (para uma mensagem iniciada pelo mestre) antes de fazer a coleta de uma resposta da estação escrava. Escolha um tempo que seja, no mínimo, igual ao tempo mais longo que uma estação escrava precisa para formatar um pacote de resposta. Geralmente, é o tempo de varredura máximo da estação escrava.	5
Polling Mode	<p>Se você deseja:</p> <ul style="list-style-type: none"> • Receba apenas uma mensagem de uma estação escrava por sua vez, escolha STANDARD (SINGLE MESSAGE TRANSFER PER NODE SCAN). Escolha este método somente se for crítico manter o tempo de varredura da lista de coleta em um mínimo. • Todas mensagens que uma estação escrava tiver, escolha STANDARD (MULTIPLE MESSAGE TRANSFER PER NODE SCAN). • Aceitar mensagens não solicitadas de estações escravas, escolha MESSAGE BASED (ALLOW SLAVES TO INITIATE MESSAGES) • As mensagens iniciadas por estação escrava são reconhecidas e processadas após todas as mensagens iniciadas por estação mestre (solicitadas). Observação: as estações escravas somente podem enviar mensagens quando forem consultadas. Se a estação mestre baseada em mensagem nunca enviar uma mensagem para uma estação escrava, a estação mestre nunca enviará uma pesquisa para a estação escrava. Portanto, para obter regularmente uma mensagem iniciada pela estação escrava de uma estação escrava, você deve escolher usar o modo de comunicação padrão. • Ignorar mensagens não solicitadas de estações escravas, escolha MESSAGE BASED (DO NOT ALLOW SLAVES TO INITIATE MESSAGES) As mensagens iniciadas pela estação escrava são reconhecidas e descartadas. A estação mestre reconhece a mensagem iniciada pela estação escrava para que a estação escrava remova a mensagem de sua fila de transmissão, o que permite que o próximo pacote programado seja transmitido para a fila de transmissão 	MESSAGE BASED (ALLOW SLAVES TO INITIATE MESSAGES)

(1) Para obter mais informações sobre o protocolo DF1, consulte [Conexão a redes usando DF1 na página 335](#).

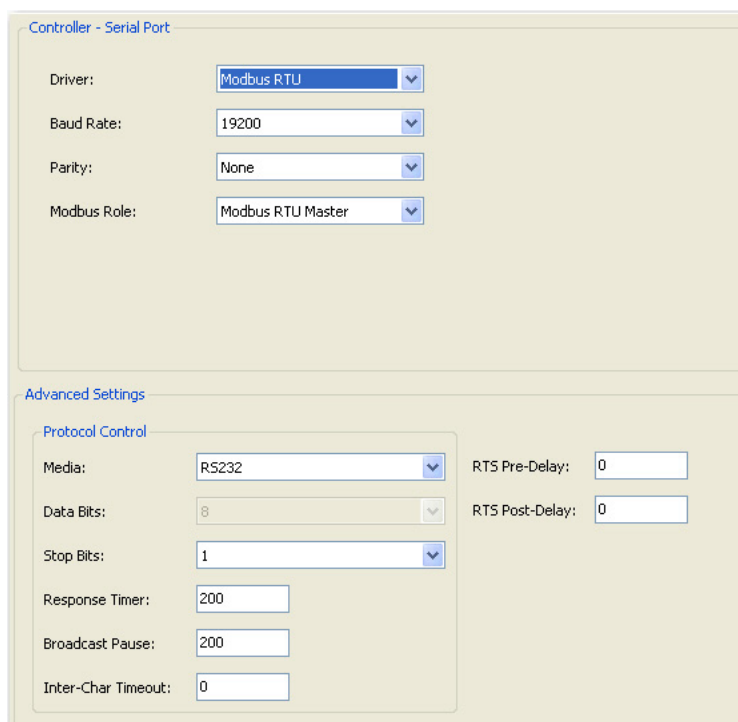
(2) Os modos half-duplex e modem de rádio DF1 são suportados somente nos controladores Micro850 (2080-L50E) e Micro870 (2080-L70E).

Configuração Modbus RTU

1. Abra o seu projeto software Connected Components Workbench. Na árvore de configuração do dispositivo, vá até Controller properties. Clique em Serial Port.



2. Selecione Modbus RTU no campo Driver.



3. Especifique os parâmetros a seguir:
 - Baud rate
 - Paridade
 - Unit address
 - Modbus Role (Master, Slave, Auto)

Parâmetros Modbus RTU

Parâmetro	Opções	Padrão
Baud Rate	1200, 2400, 4800, 9600, 19200, 38400	19200
Paridade	None, Odd, Even	Nenhuma
Modbus Role	Master, Slave, Auto	Mestre

4. Clique em Advanced Settings para definir os parâmetros avançados. Consulte a tabela para as opções disponíveis e configuração padrão para os parâmetros avançados.

Parâmetros avançados Modbus RTU

Parâmetro	Opções	Padrão
Media	RS-232, RS-232 RTS/CTS, RS-485	RS-232
Bits de dados	Sempre 8	8
Bits de parada	1, 2	1
Response timer	0 a 999.999.999 milissegundos	200
Broadcast Pause	0 a 999.999.999 milissegundos	200
Inter-char timeout	0 a 999.999.999 microssegundos	0
RTS Pre-delay	0 a 999.999.999 microssegundos	0
RTS Post-delay	0 a 999.999.999 microssegundos	0

Configurar ASCII

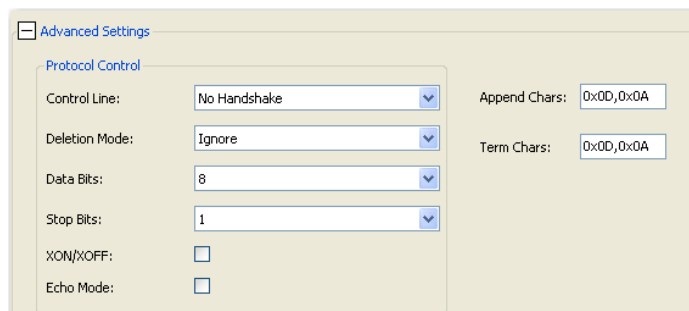
1. Abra o seu projeto software Connected Components Workbench. Na árvore de configuração do dispositivo, vá até Controller properties. Clique em Serial Port.
2. Selecione ASCII no campo Driver.

3. Especifique a velocidade de transmissão e a paridade.

Parâmetros ASCII

Parâmetro	Opções	Padrão
Baud Rate	1200, 2400, 4800, 9600, 19200, 38400	19200
Paridade	None, Odd, Even	Nenhuma

4. Clique em Advanced Settings para definir os parâmetros avançados.

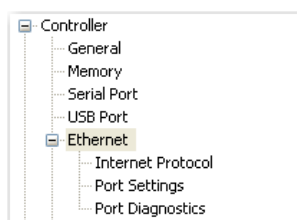


Parâmetros avançados ASCII

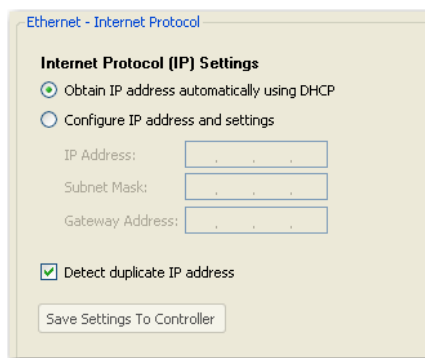
Parâmetros	Opções	Padrão
Control Line	Full Duplex Half-duplex com portadora contínua Half-duplex sem portadora contínua (RTS/CTS) No Handshake	No Handshake
Deletion Mode	CRT Ignorar Impressora	Ignorar
Bits de dados	7, 8	8
Bits de parada	1, 2	1
XON/XOFF	Habilitado ou desabilitado	Desabilitado
Echo Mode	Habilitado ou desabilitado	Desabilitado
Append Chars	0x0D,0x0A ou valor especificado pelo usuário	0x0D,0x0A
Term Chars	0x0D,0x0A ou valor especificado pelo usuário	0x0D,0x0A

Configurar as definições de Ethernet

1. Abra o seu projeto software Connected Components Workbench (por exemplo, Micro850). Na árvore de configuração do dispositivo, vá até Controller properties. Clique em Ethernet.



2. Sob Ethernet, clique em Internet Protocol.
Configure os parâmetros de protocolo de internet (IP). Especifique se deseja obter o endereço IP automaticamente utilizando DHCP ou configurar manualmente o endereço IP, máscara de subrede e endereço de conversor de protocolos.

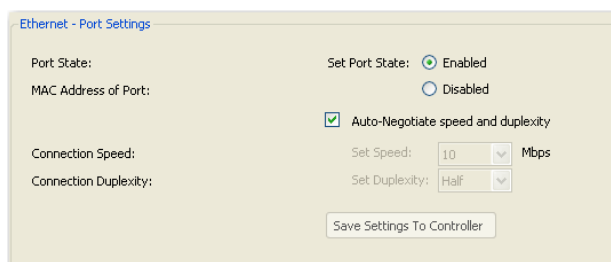


A porta Ethernet assume como padrão de fábrica as seguintes configurações:

- DHCP (endereço IP dinâmico)
- Detecção de endereço duplicado: Ligada

IMPORTANTE Quando um servidor DHCP falha, o controlador Micro800 aloca endereços IP no intervalo particular 169.254.0.1 a 169.254.255.254. O controlador Micro800 verifica se seu endereço é único na rede usando ARP. Quando o servidor DHCP for novamente capaz de atender solicitações, o controlador Micro800 atualizará seu endereço automaticamente.

3. Clique na caixa de marcar Detect duplicate IP address para habilitar a detecção de endereço duplicado.
4. Em Ethernet, clique em Definições da porta.



5. Defina Port State como habilitado ou desabilitado.
6. Para definir manualmente a velocidade e duplicidade da conexão, desmarque a caixa de opção Auto-Negotiate speed e duplexity. Então, defina os valores para Speed (10 ou 100 Mbps) e Duplexity, valores (Half ou Full).
7. Clique em Save Settings to Controller se você deseja salvar as configurações para o seu controlador.
8. Na árvore de configuração do dispositivo, sob Ethernet, clique em Port Diagnostics para monitorar contadores de mídia e interface. Os contadores estão disponíveis e atualizados quando o controlador está em modo de depuração.

Validar endereço IP

Os módulos devem validar a configuração do endereço IP de entrada, independentemente de ele ter sido obtido por configuração explícita ou por DHCP.

As regras a seguir devem ser obedecidas ao configurar o endereço IP:

- O endereço IP para o módulo não pode ser definido como zero, um endereço multicast, um endereço de transmissão nem um endereço na rede de malha reversa classe A (127.x.x.x).
- O endereço IP não deve começar com zero, e a ID da rede do endereço IP não deve ser zero.
- A máscara de rede não pode ser definida como 255.255.255.255.
- O endereço do conversor de protocolos deve estar na mesma sub-rede que o endereço IP que está sendo configurado.
- O endereço do servidor de nome não pode ser definido como zero, um endereço multicast, um endereço de transmissão nem um endereço na rede de malha reversa classe A (127.x.x.x).

A faixa válida de endereço IP estático IPv4 exclui:

- IP de transmissão ou zero (255.255.255.255 ou 0.0.0.0)
- Endereço IP começando com 0 ou 127 (0.xxx.xxx.xxx ou 127.xxx.xxx.xxx)
- Endereço IP terminando com 0 ou 255 (xxx.xxx.xxx.0 ou xxx.xxx.xxx.255)
- Endereços IP no intervalo 169.254.xxx.xxx (169.254.0.0 a 169.254.255.255)
- Endereços IP no intervalo 224.0.0.0 a 255.255.255.255

Nome do host Ethernet

Os controladores Micro800 implementam nomes de host exclusivo a cada controlador, para serem usados para identificar o controlador na rede. O nome do host padrão é composto por duas partes: tipo do produto e endereço MAC, separados por um hífen. Por exemplo: 2080LC50-xxxxxxxxxxxx, onde xxxxxxxxxxxx é o endereço MAC.

O usuário pode alterar o nome do host usando o CIP Service Set Attribute Single quando o controlador está no modo Programa/Modo de programa remoto.

Configurar o driver serial CIP

1. Abra o seu projeto do software Connected Components Workbench. Na árvore de configuração do dispositivo, vá até Controller properties. Clique em Serial Port.
2. Selecione CIP Serial a partir do campo Driver.
3. Especifique uma baud rate. Selecione uma taxa de comunicação que seja suportada por todos os dispositivos no seu sistema. Configure todos os dispositivos no sistema para a mesma taxa de comunicação. A baud rate padrão é definida como 38.400 bps.
4. Na maioria dos casos, os parâmetros parity e station address devem ser deixados nas suas configurações padrão.
5. Clique em Advanced Settings e defina os parâmetros avançados.

Suporte OPC usando FactoryTalk Linx

O suporte para Open Platform Communications (OPC) usando CIP symbolic foi adicionado a partir do firmware release 7.0 e posteriores. Ele pode ser usado no lugar do endereçamento Modbus.

São necessários RSLinx Enterprise versão 5.70 (CPR9 SR7) ou posterior e FactoryTalk Gateway versão 3.70 (CPR9 SR7) ou posterior.

Observações:

Protocolo de rede distribuída do controlador Micro870

Visão geral

Este capítulo descreve como definir as configurações de comunicação do DNP3. Os temas incluem:

Tópico	Página
Configuração de canal para escravo DNP3	77
Camada de aplicação escrava DNP3	103
Objetos DNP3 e variáveis do controlador	108
Objeto de atributo de dispositivo DNP3	119
Relatório de eventos	121
Prevenção de colisão	126
Sincronização de Tempo	126
Diagnóstico	128
Códigos de função	130
Tabela de implementação	131

Configuração de canal para escravo DNP3

O protocolo de comunicação padrão para as portas seriais é DF1 Full-Duplex. Para comunicar-se com o protocolo de rede distribuída (DNP3), o canal deve ser configurado para escravo DNP3.

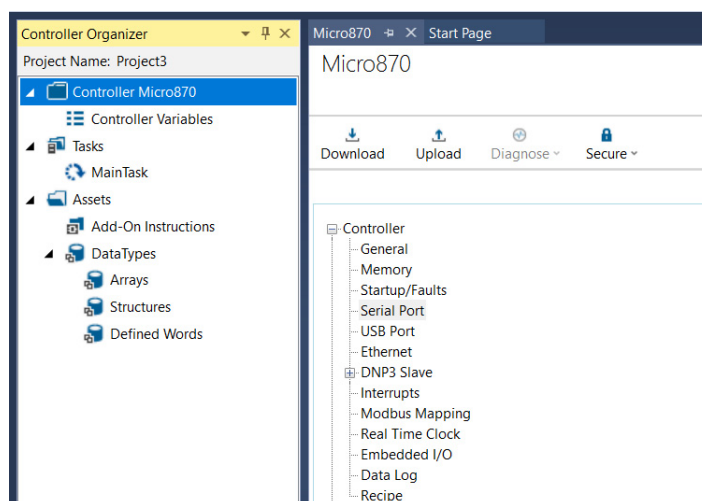
O protocolo de comunicação padrão para o canal Ethernet no controlador é EtherNet/IP. Para se comunicar com o protocolo DNP3 sobre IP, selecione DNP3 over IP Enable na página de configuração Ethernet.

IMPORTANTE O protocolo DNP3 é suportado somente nos seguintes controladores Micro870.

- 2080-L70E-24QBBN
- 20080-L70E-24QWBN

Para programar o controlador, use o software Connected Components Workbench versão 20.01.00 ou posterior.

No software Connected Components Workbench, abra a árvore de projeto do controlador Micro870.



Existem três configurações relativas ao protocolo DNP3 no software Connected Components Workbench:

- Configuração de porta serial e/ou Ethernet
- Configuração da camada de aplicação escrava DNP3.

Configuração da camada do link da porta serial

A configuração relativa à camada de link pode ser feita na página Serial Port configuration.

Controller - Serial Port

Driver: DNP3 Slave +

Baud Rate: 38400

Parity: None

Node Address: 1

Protocol Control

☒ Enable Master Address Validation

☐ Enable Self-Address

Master Node 0: 0 Master Node 1: 0 Master Node 2: 0

Master Node 3: 0 Master Node 4: 0

Media: RS485

Control Line: No Handshake

Confirmation Timeout: 20 ms

Stop Bits: 1

☐ Request LL Confirmation

☐ Send LL Confirmation

Message Retries: 0 Max Random Delay: 0 ms

Pre Transmit Delay: 0 ms

Configuração da camada Ethernet

Para habilitar o protocolo DNP3 sobre IP, selecione DNP3 over IP Enable na página Ethernet configuration.

The screenshot shows the Micro870 configuration interface. On the left is a tree view with categories like Controller, Motion, Plug-in Modules, and Expansion Modules. The 'Ethernet' option under the 'Controller' category is selected. On the right, the 'Controller - Ethernet' configuration page is displayed. It includes settings for Port State, Auto-Negotiate Speed and Duplex Mode, Internet Protocol (IP) Settings (DHCP or manual IP), EtherNet/IP Inactivity Timeout, Modbus TCP Server state, and the 'DNP3 Over IP' section. The 'DNP3 over IP Enable' checkbox is highlighted with a red box.

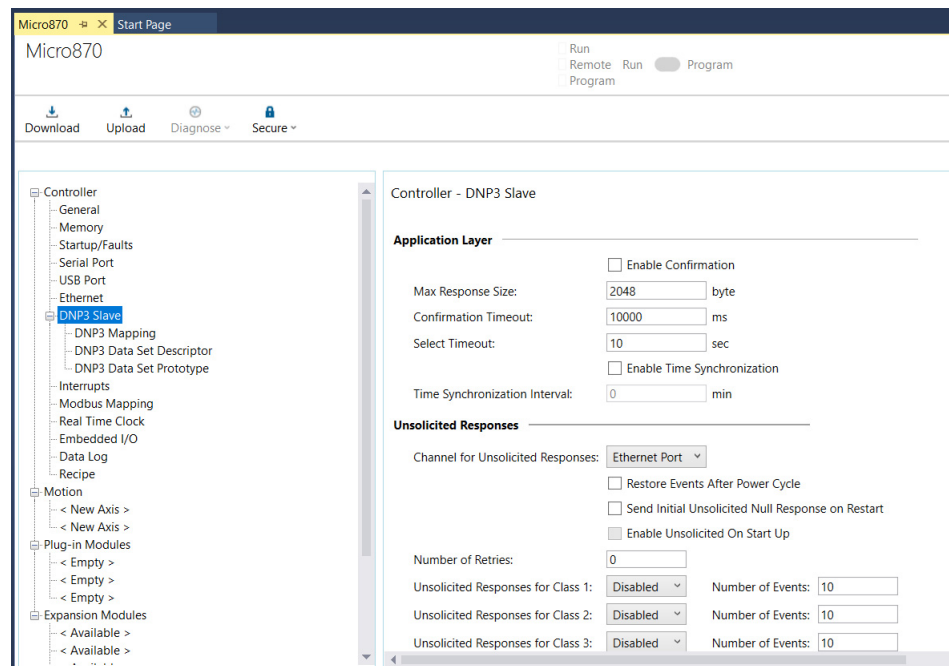
A configuração relativa à camada de link pode ser feita na página Ethernet configuration.

The screenshot shows the 'Controller - Ethernet' configuration page with the 'DNP3 Over IP' section expanded. It contains the following settings:

- ☒ DNP3 over IP Enable
- Slave Node Address: 1
- ☒ Enable Master Address Validation
- ☐ Enable Self-Address
- Master Node 0: 0, Master Node 1: 0, Master Node 2: 0
- Master Node 3: 0, Master Node 4: 0
- ☒ Enable Access Control for Master IP Address
- Master IP 0: 0 . 0 . 0 . 0
- Master IP 1: 0 . 0 . 0 . 0
- Master IP 2: 0 . 0 . 0 . 0
- Master IP 3: 0 . 0 . 0 . 0
- Master IP 4: 0 . 0 . 0 . 0
- End Point Type: Listening
- Local Port Number(UDP): 20000
- Local Port Number(TCP): 20000
- Keep Alive Interval: 10 sec

Configuração da camada de aplicação escrava DNP3

A configuração relativa à camada de aplicação pode ser feita na página DNP3 Slave configuration.



A configuração do escravo DNP3 é compartilhada pelas portas serial e Ethernet se várias portas forem configuradas para o protocolo DNP3. Qualquer alteração na página de configuração do escravo DNP3 afeta todas as portas.

Parâmetros de configuração da camada do link serial

Driver

Esta seleção deve ser definida como escravo DNP3 para se comunicar com o protocolo DNP3.

Endereço do nó

Este valor é um endereço do nó deste escravo DNP3.
A faixa válida é de 0 a 65519.
O valor padrão é 1.

Baud

As seleções podem ser 38400, 19200, 9600, 4800, 2400 e 1200.
A seleção padrão é 38400.

Paridade

As seleções podem ser NONE, EVEN e ODD. A seleção padrão é NONE.

Bits de parada

As seleções podem ser 1 e 2.
A seleção padrão é 1.

Habilitar validação do endereço mestre

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está desabilitada (desmarcada), o controlador aceita as solicitações de qualquer DNP3 mestre.

Quando a seleção está habilitada (marcada), o controlador aceita as solicitações apenas do DNP3 mestre que estiver configurado no Mestre Node0 a Node4. O número máximo de endereços do nó mestre para a validação do endereço mestre é 5.

Habilitar autoendereçoamento

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando este bit está desabilitado (desmarcado), qualquer pacote que contenha o endereço de destino 65532 (FFFCh) será ignorado.

Quando este bit está habilitado (marcado), qualquer pacote que contenha o endereço de destino 65532 (FFFCh) será aceito e processado.

Qualquer resposta de volta ao DNP3 mestre contém o endereço DNP3 configurado real do controlador Micro870.

Mestre Node0

Este valor é usado para:

- validar o endereço do nó mestre quando Habilitar validação do endereço mestre estiver habilitado (marcado)
- enviar resposta não solicitada quando a funcionalidade de resposta não solicitada estiver habilitada. Uma resposta não solicitada é enviada ao mestre DNP3 que tem este endereço.

A faixa válida é de 0 a 65.519. O valor padrão é 0.

Mestre Node1, Mestre Node2, Mestre Node3, Mestre Node4

A faixa válida é de 0 a 65.519. O valor padrão é 0.

Esse valor é usado para verificar a validação do endereço do nó mestre quando Habilitar validação do endereço mestre estiver habilitada (marcada).

Linha de controle

As seleções podem ser Sem reconhecimento e Half-Duplex sem portadora contínua (CTS/RTS). A seleção padrão é Sem reconhecimento.

Quando o controlador estiver conectado ao DNP3 mestre usando a linha RS-232 diretamente, você deve selecionar Sem reconhecimento. Se quiser usar a linha do modem em uma rede half-duplex, você deve selecionar Half-duplex sem portadora contínua (CTS/RTS). Se o controlador estiver conectado a uma rede RS-485, você deve selecionar sem reconhecimento.

Solicitar confirmação LL

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção é Desabilitada (desmarcada), os frames primários do controlador são enviados com o código de função FC_UNCONFIRMED_USER_DATA (4).

Quando a seleção é Habilitada (marcada), os frames primários do controlador são enviados com o código de função FC_CONFIRMED_USER_DATA (3). Neste caso, o controlador aguarda a confirmação e pode tentar novamente o frame se não receber a confirmação do mestre DNP3 dentro do tempo limite de confirmação (x1 ms).

Enviar confirmação LL

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está desabilitada (desmarcada), o frame secundário opcional não é enviado com o código de função FC_NACK (1) ou FC_NOT_SUPPORTED (15).

Quando a seleção está habilitada (marcada), o frame secundário opcional é enviado com o código de função FC_NACK (1) ou FC_NOT_SUPPORTED (15).

IMPORTANTE Os controladores Micro870 (2080-L70E-24QxBN) suportam esta função somente quando o DNP3 mestre envia dados de usuário confirmados. Esta função não é suportada quando o mestre DNP3 envia dados de usuário não confirmados.

Tempo limite de confirmação (x1 ms)

Quando Request LL Confirmation está habilitado, o controlador aguarda para receber um quadro de confirmação até que este tempo limite expire.

A faixa válida é de 1 a 65535. O valor padrão é 20.

Nova tentativa de mensagem

Quando o tempo-limite de confirmação (x1 ms) expira e este parâmetro era um valor diferente de zero, o controlador tenta enviar pacotes de nova tentativa.

A faixa válida é de 1 a 255. O valor padrão é 0.

Atraso de pré-transmissão (x1 ms)

O controlador aguarda o tempo especificado antes de enviar o pacote.

A faixa válida é de 0 a 65535. O valor padrão é 0.

Atraso na desenergização de RTS (x1 ms)

Quando Control estiver definido como half-duplex Modem (CTS/RTS handshaking), esse recurso será habilitado. Especifica um atraso de tempo entre o fim de uma transmissão e a queda do sinal RTS.

A faixa válida é de 0 a 65535. O valor padrão é 0.

Atraso de envio RTS (x1 ms)

Quando Control estiver definido como Half-duplex Modem (CTS/RTS handshaking), essa entrada será habilitada. Isso especifica um atraso de tempo entre a elevação do RTS e o início de uma transmissão.

A faixa válida é de 0 a 65535. O valor padrão é 0.

Atraso máximo aleatório (x1 ms)

Este parâmetro é usado com o atraso de pré-transmissão (x1 ms) para prevenção de colisão na rede RS-485. Para obter mais detalhes, consulte [Prevenção de colisão na página 126](#).

A faixa válida é de 0 a 65535. O valor padrão é 0.

Parâmetros de configuração da camada Ethernet

O subsistema DNP3 sobre IP no controlador suporta o ponto de extremidade de escuta, ponto de extremidade TCP duplo e tipo de ponto de extremidade de datagrama.

O tipo de ponto de extremidade de escuta suporta uma única conexão TCP como um servidor e um datagrama UDP.

O tipo TCP Dual End Point suporta uma única conexão TCP como um servidor, uma única conexão TCP como um cliente e um datagrama UDP.

O tipo de ponto de extremidade de datagrama suporta o datagrama UDP dos mestres DNP3. Os números de porta TCP e UDP padrão são 20000 e os números de porta são configuráveis.

O tipo de ponto de extremidade pode ser determinado pelo parâmetro End Point Type. De acordo com o parâmetro, o controlador funciona como diferentes tipos de ponto de extremidade. Consulte a [Tabela 6](#) para cada configuração.

Tabela 6 – Tipos de ponto de extremidade

Tipo de ponto de extremidade	Conexão	Descrição
Ponto de extremidade de escuta	Uma única conexão de servidor TCP	Qualquer uma das solicitações é aceita e as respostas são transmitidas por esta conexão. As respostas não solicitadas são transmitidas por esta conexão quando esta conexão está disponível.
	datagrama UDP	Aceita apenas pacotes de transmissão quando o nó de destino DNP3 for um dos 0xFFFFD, 0xFFFFE e 0xFFFF na solicitação.
Ponto de extremidade duplo	Uma única conexão de servidor TCP	Qualquer uma das solicitações é aceita e as respostas são transmitidas por esta conexão. As respostas não solicitadas são transmitidas por esta conexão quando esta conexão está disponível. Esta conexão tem prioridade mais alta que a conexão do cliente.
	Uma única conexão de cliente TCP	Qualquer uma das solicitações é aceita e as respostas são transmitidas por esta conexão. As respostas não solicitadas são transmitidas por esta conexão quando esta conexão está disponível. O controlador não solicita a conexão do cliente TCP com o DNP3 mestre até que uma resposta não solicitada seja gerada.
	datagrama UDP	Aceita apenas pacotes de transmissão quando o nó de destino DNP3 for um dos 0xFFFFD, 0xFFFFE e 0xFFFF na solicitação.
Ponto de extremidade do datagrama	Apenas datagrama UDP	Qualquer uma das solicitações é aceita e as respostas são transmitidas por esta conexão. Todas as respostas podem ser transmitidas para a porta DNP3 mestre diferente de acordo com a configuração dos parâmetros Remote UDP Port Number e Master IP Address0. Se este parâmetro não for definido como 0, as respostas solicitadas serão enviadas para a porta DNP3 mestre que está configurada. Se este parâmetro for definido como 0, as respostas solicitadas serão enviadas para a porta DNP3 mestre que enviou a solicitação. A conexão TCP não está disponível nesta configuração.

O parâmetro DNP3 over IP Enable é configurado na página Ethernet configuration e outros parâmetros são configurados na página DNP3 Slave configuration.

DNP3 over IP Enable

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção estiver desabilitada (desmarcada), o serviço DNP3 por Ethernet é desabilitado após a configuração ser baixada para o controlador.

Quando a seleção estiver habilitada (marcada), o serviço DNP3 por Ethernet é habilitado após a configuração ser baixada para o controlador.

Habilitar validação do endereço mestre

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está desabilitada (desmarcada), o controlador aceita as solicitações de qualquer DNP3 mestre.

Quando a seleção está habilitada (marcada), o controlador aceita apenas as solicitações do endereço do nó DNP3 mestre, que é configurado nos parâmetros [Mestre Node0 na página 81](#) e [Mestre Node1, Mestre Node2, Mestre Node3, Mestre Node4 na página 81](#). O número máximo de endereços do nó mestre para a validação do endereço mestre é 5.

Habilitar autoendereçoamento

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando este bit está desabilitado (desmarcado), qualquer pacote que contenha o endereço de destino 65532 (FFFCh) será ignorado.

Quando este bit está habilitado (marcado), qualquer pacote que contenha o endereço de destino 65532 (FFFCh) será aceito e processado.

Qualquer resposta de volta ao DNP3 mestre contém o endereço DNP3 configurado real do controlador Micro870.

Habilitar controle de acesso

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está desabilitada (desmarcada), o controlador aceita as solicitações de qualquer DNP3 mestre.

Quando a seleção está habilitada (marcada), o controlador aceita as solicitações apenas do endereço IP DNP3 mestre, que é configurado nos parâmetros Master IP Address0 a Master IP Address4. O número máximo de endereços IP do mestre para o controle de acesso é 5.

Tipo de ponto de extremidade

As seleções válidas são Listening, Dual e Datagram Only.

O padrão é tipo de extremidade Listening.

Mestre Node0

Este valor é usado para:

- validar o endereço do nó mestre quando Habilitar validação do endereço mestre estiver habilitado (marcado)

- enviar resposta não solicitada quando a funcionalidade de resposta não solicitada estiver habilitada. Uma resposta não solicitada é enviada ao mestre DNP3 que tem este endereço.

A faixa válida é de 0 a 65519. O valor padrão é 0.

Mestre Node1, Mestre Node2, Mestre Node3, Mestre Node4

Esse valor é usado para validação do endereço do nó mestre quando Habilitar validação do endereço mestre estiver habilitada (marcado). Este valor é mostrado e válido somente quando Habilitar validação de endereço mestre estiver habilitado (marcado).

A faixa válida é de 0 a 65519. O valor padrão é 0.

IP do mestre Address0

Este valor é usado para:

- validar o endereço IP do mestre quando Habilitar controle de acesso estiver habilitado (marcado)
- enviar resposta não solicitada quando a funcionalidade de resposta não solicitada estiver habilitada. Uma resposta não solicitada é enviada ao mestre DNP3 que tem este endereço.

O valor válido é um endereço IP. O valor padrão é 0.0.0.0.

IP do mestre Address1, IP do mestre Address2, IP do mestre Address3, IP do mestre Address4

Esse valor é usado para validação do endereço IP do mestre quando Habilitar validação de controle de acesso estiver habilitado (marcado). Este valor somente é mostrado e válido quando Habilitar controle de acesso estiver habilitado (marcado).

O valor válido é um endereço IP. O valor padrão é 0.0.0.0.

Núm. da Porta TCP mestre (Não sol.)

Este valor é usado para configurar o Núm. da Porta TCP mestre para resposta não solicitada.

A faixa válida é de 0 a 65535. O valor padrão é 20.000.

DNP3 Over IP

☒ DNP3 over IP Enable

Slave Node Address:

☒ Enable Master Address Validation

☒ Enable Self-Address

Master Node 0: Master Node 1: Master Node 2:

Master Node 3: Master Node 4:

☐ Enable Access Control for Master IP Address

Master IP 0:

End Point Type:

Local Port Number(UDP):

Local Port Number(TCP):

Keep Alive Interval: sec

Master TCP Port Number(Unsol):

Núm. da porta UDP mestre para inicial não solicitada

Este valor é usado para configurar o número de porta UDP mestre para resposta inicial não solicitada se o parâmetro tipo de ponto de extremidade estiver selecionado como Datagram Only.

A faixa válida é de 0 a 65535. O valor padrão é 20.000.

Núm. da Porta UDP mestre

Este valor é usado para configurar o Núm. da Porta UDP mestre para resposta inicial não solicitada se o parâmetro tipo de ponto de extremidade estiver selecionado como Datagram Only.

A faixa válida é de 0 a 65535. O valor padrão é 20.000.

DNP3 Over IP

☒ DNP3 over IP Enable

Slave Node Address:

☒ Enable Master Address Validation

☒ Enable Self-Address

Master Node 0: Master Node 1: Master Node 2:

Master Node 3: Master Node 4:

☐ Enable Access Control for Master IP Address

Master IP 0:

End Point Type:

Local Port Number(UDP):

Master UDP Port Number(Unsol):

Master UDP Port Number(Init Unsol):

Intervalo de Keep Alive (x 1s)

Este parâmetro especifica um intervalo de tempo para o mecanismo TCP Keep Alive.

Se o temporizador atingir o tempo-limite, o controlador transmite uma mensagem de keep-alive. A mensagem de keep-alive é uma solicitação de status da camada de enlace de dados DNP (FC_REQUEST_LINK_STATUS). Se não for recebida uma resposta para a mensagem de keep-alive, o controlador considera a conexão TCP interrompida e a termina.

A faixa válida é de 1 a 65535. O valor padrão é 10.

Endereço do nó escravo

Este valor é um endereço do nó deste escravo DNP3.

A faixa válida é de 0 a 65519. O valor padrão é 1.

Núm. da Porta local (UDP)

Este valor é usado para configurar o Núm. da Porta UDP local que é usado para escuta de soquete UDP.

A faixa válida é de 0 a 65535. O valor padrão é 20.000.

Núm.da Porta local (TCP)

Este valor é usado para configurar o Núm. da Porta TCP que é usado para escuta de soquete TCP.

A faixa válida é de 0 a 65535. O valor padrão é 20.000.

DNP3 Over IP

☒ DNP3 over IP Enable

Slave Node Address: 1

☒ Enable Master Address Validation

☒ Enable Self-Address

Master Node 0: 0 Master Node 1: 0 Master Node 2: 0

Master Node 3: 0 Master Node 4: 0

☐ Enable Access Control for Master IP Address

Master IP 0: 0, 0, 0, 0

End Point Type: Listening

Local Port Number(UDP): 20000

Local Port Number(TCP): 20000

Keep Alive Interval: 10 sec

Parâmetros de configuração da camada de aplicação escrava DNP3*Canal para resposta não solicitada*

Somente os canais já configurados para o protocolo DNP3 aparecem no menu suspenso Channel for Unsolicited Response. Todas e quaisquer respostas não solicitadas são transmitidas por meio deste canal selecionado.

Restaurar eventos após desligar e ligar a alimentação

Quando a seleção está Desabilitada (desmarcada), os eventos DNP3 gerados antes de desligar a alimentação são descartados após ligar a alimentação novamente. Quando a opção está Habilitada (marcada), todos os eventos DNP3 serão restaurados após desligar e ligar a alimentação.

As seleções válidas são Enabled (marcada) e Disabled (desmarcada), com valor padrão desabilitado.

Respostas não solicitadas na inicialização

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Após um reinício, quando a seleção está Desabilitada (desmarcada), o controlador só enviará respostas não solicitadas habilitadas quando receber um comando FC_ENABLE_UN SOLICITED (20) do DNP3 mestre.

Após um reinício, quando a seleção está Habilitada (marcada), o controlador envia todas as respostas não solicitadas habilitadas para o mestre DNP3 incondicionalmente.

Respostas não solicitadas para Classe 1

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está Desabilitada (desmarcada), a Resposta não solicitada é desabilitada para eventos de Classe 1. Para evitar overflow do buffer de eventos, o DNP3 mestre deve coletar eventos de Classe 1.

Quando a seleção está Habilitada (marcada), a Resposta não solicitada é habilitada para eventos de Classe 1.

Respostas não solicitadas para Classe 2

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está Desabilitada (desmarcada), a Resposta não solicitada é desabilitada para eventos de Classe 2. Para evitar overflow do buffer de eventos, o DNP3 mestre deve coletar eventos de Classe 2.

Quando a seleção está Habilitada (marcada), a Resposta não solicitada é habilitada para eventos de Classe 2.

Habilitar não solicitado para Classe 3

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está Desabilitada (desmarcada), a Resposta não solicitada é desabilitada para eventos de Classe 3. Para evitar overflow do buffer de eventos, o DNP3 mestre deve coletar eventos de Classe 3.

Quando a seleção está Habilitada (marcada), a Resposta não solicitada é habilitada para eventos de Classe 3.

Enviar resposta nula não solicitada inicial na inicialização

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está desabilitada (desmarcada), o controlador não envia resposta NULL não solicitada com o bit RESTART IIN na inicialização.

Quando a seleção está habilitada (marcada), o controlador envia resposta NULL não solicitada com o bit RESTART IIN na inicialização.

Esta seleção também é usada para enviar o bit Restart IIN durante as alterações de configuração de driver e de canal. [Consulte Indicações internas na página 107](#) para obter mais informações.

Habilitar confirmação

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está desabilitada (desmarcada), o controlador envia pacotes de resposta com bit CON definido em seu cabeçalho somente sob as condições a seguir:

- Quando a resposta tem dados de evento.
- Quando a resposta é de múltiplos fragmentos.
- Quando a resposta não solicitada é enviada.

Quando a seleção está habilitada (marcada), o controlador sempre envia pacotes de resposta com o bit CON definido em seu cabeçalho, o que faz com que o mestre DNP3 envie respostas confirmando que recebeu cada pacote de resposta sem erro.

Habilite a sincronização de tempo

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Este parâmetro é usado com o intervalo de sincronização de tempo (x1 min).

Quando a seleção está desabilitada (desmarcada), o controlador não realiza nenhuma sincronização de tempo.

Quando a seleção está habilitada (marcada), o controlador define o bit de indicação interna NEED_TIME (IIN1.4) na energização e em cada intervalo configurado no intervalo de sincronização de tempo (x1 min).

Intervalo de sincronização de tempo (x1 min)

Este parâmetro é usado com habilitar sincronização de tempo. Válido somente quando Habilitar sincronização de tempo estiver habilitado (marcado).

A faixa válida é de 0 a 32767. O valor padrão é 0. Se o valor for 0, o bit de indicação interna NEED_TIME (IIN1.4) é definido na inicialização e depois a cada minuto do intervalo de sincronização de tempo se o valor for maior que 0.

Quando o parâmetro Enable Time Synchronization está desabilitado (desmarcado), o bit IIN1.4 nunca é ligado.

Tamanho máx. de resposta

O controlador envia o frame da camada da aplicação para ajustar ao tamanho máximo máx. de resposta. Se o tamanho do pacote de resposta for maior que esse valor, o controlador fragmenta o pacote de resposta.

A faixa válida é de 27 a 2048 em bytes. O valor padrão é 2048.

Tempo limite de confirmação (x1 ms)

Quando Habilitar confirmação está habilitado, o controlador aguarda a Confirmação da camada da aplicação até que tempo limite de confirmação (x1 ms) expire.

A faixa válida é de 100 a 65535 com incrementos de 1 ms. O valor padrão é 10000.

Número de novas tentativas

Este parâmetro é apenas para resposta não solicitada. Se este valor tiver o máximo, 65535, significa tentativas infinitas de resposta não solicitada.

A faixa válida é de 0 a 65535. O valor padrão é 0.

Número de eventos Classe 1

Se o controlador estiver configurado para não iniciar respostas não solicitadas de classe 1, este parâmetro é usado para limitar o número máximo de eventos que são gerados e registrados no buffer de eventos para eventos de Classe 1. Neste caso, o valor 0 desabilita a geração de eventos de Classe 1.

Se o controlador estiver configurado para gerar respostas não solicitadas e o número de eventos de Classe 1 enfileirados atingir esse valor, é iniciada uma resposta não solicitada.

A faixa válida é de 0 a 10000. O valor padrão é 10.

Consulte [Registro de eventos DNP3 10K na página 123](#) para obter mais informações.

Tempo de retenção após eventos de Classe 1 (x 1s)

Este parâmetro é apenas para resposta não solicitada de Classe 1. O controlador retém os eventos durante o tempo de retenção após eventos de Classe 1 (x 1s) antes de iniciar uma resposta não solicitada.

A faixa válida é de 0 a 65535. O valor padrão é 5.

O valor 0 indica que as respostas não serão atrasadas devido a este parâmetro.

Os parâmetros Number of Classe 1 Events e Hold Time after Classe 1 Events (x 1s) são usados juntos para que, se um dos critérios for atendido, seja transmitida uma resposta não solicitada.

Por padrão, o tempo de retenção é disparado novamente para cada novo evento detectado.

Número de eventos Classe 2

Se o controlador estiver configurado para não iniciar respostas não solicitadas de classe 2, este parâmetro é usado para limitar o número máximo de eventos que são gerados e registrados no buffer de eventos para eventos de Classe 2. Neste caso, o valor 0 desabilita a geração de eventos de Classe 2.

Se o controlador estiver configurado para gerar respostas não solicitadas e o número de eventos de Classe 2 enfileirados atingir esse valor, é iniciada uma resposta não solicitada.

A faixa válida é de 0 a 10000. O valor padrão é 10.

Consulte [Registro de eventos DNP3 10K na página 123](#) para obter mais informações

Tempo de retenção após eventos de Classe 2 (x 1s)

Este parâmetro é apenas para resposta não solicitada de Classe 2.

O controlador retém os eventos durante o tempo de retenção após eventos de Classe 2 (x 1s) antes de iniciar uma resposta não solicitada.

A faixa válida é de 0 a 65535. O valor padrão é 5.

O valor 0 indica que as respostas não serão atrasadas devido a este parâmetro.

Os parâmetros Number of Classe 2 Events e Hold Time after Classe 2 Events (x 1s) são usados juntos para que, se um dos critérios for atendido, seja transmitida uma resposta não solicitada.

Por padrão, o tempo de retenção é disparado novamente para cada novo evento detectado.

Número de eventos Classe 3

Se o controlador estiver configurado para não iniciar respostas não solicitadas de classe 3, este parâmetro é usado para limitar o número máximo de eventos que são gerados e registrados no buffer de eventos para eventos de Classe 3. Neste caso, o valor 0 desabilita a geração de eventos de Classe 3.

Se o controlador estiver configurado para gerar respostas não solicitadas e o número de eventos de Classe 3 enfileirados atingir esse valor, é iniciada uma resposta não solicitada.

A faixa válida é de 0 a 10000. O valor padrão é 10.

Consulte [Registro de eventos DNP3 10K na página 123](#) para obter mais informações.

Tempo de retenção após eventos de Classe 3 (x 1s)

Este parâmetro é apenas para resposta não solicitada de Classe 3.

O controlador retém os eventos durante o tempo de retenção após eventos de Classe 3 (x 1s) antes de iniciar uma resposta não solicitada.

A faixa válida é de 0 a 65535. O valor padrão é 5.

O valor 0 indica que as respostas não serão atrasadas devido a este parâmetro.

Os parâmetros Number of Classe 3 Events e Hold Time after Classe 3 Events (x 1s) são usados juntos para que, se um dos critérios for atendido, seja transmitida uma resposta não solicitada.

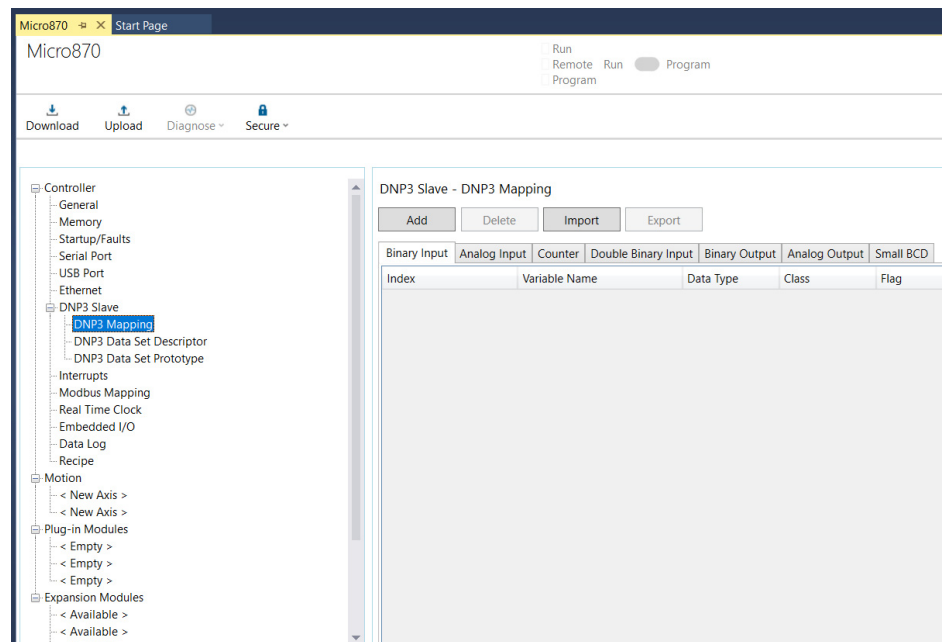
Por padrão, o tempo de retenção é disparado novamente para cada novo evento detectado.

Selecionar tempo limite (x 1s)

A faixa válida é de 1 a 65535. O valor padrão é 10.

Este parâmetro é usado para controlar o bloco de saída de relé de controle (CROB) e o bloco de saída analógica (AOB). Após receber a solicitação com o código de função FC_SELECT(3), o DNP3 mestre deve enviar a solicitação com o código de função FC_OPERATE(4) dentro deste tempo configurado.

Dados e configuração do objeto DNP3



A seleção de mapeamento DNP3 em DNP3 escravo nas propriedades do controlador permite definir o mapeamento do objeto DNP3 listado e as propriedades do objeto (número de classe, status on-line/off-line, flags de qualidade do objeto, bandas mortas e/ou limites) para variáveis do controlador.

Consulte [Objetos DNP3 e variáveis do controlador na página 108](#) para obter mais informações.

Autenticação segura DNP3

O controlador implementa a autenticação segura DNP3 com base na especificação DNP3, suplemento para o Volume 2, autenticação segura, Versões 2.00 e 5.00.

A autenticação segura DNP3 foi implementada na camada de aplicação DNP3 do sistema do controlador. Todos os parâmetros que você configurar em relação à autenticação segura DNP3 na configuração da camada de aplicação escravo DNP3 eles afetarão todas as portas configuradas para o protocolo DNP3 no controlador.

Habilitar autenticação segura

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está Desabilitada (desmarcada), o controlador desabilita o subsistema de autenticação segura DNP3.

Quando a seleção está Habilitada (marcada), o controlador habilita o subsistema de autenticação segura DNP3.

Versão de autenticação segura

Este parâmetro especifica a versão de autenticação que este controlador escravo DNP3 usa.

Selecione 2 para autenticação segura versão 2 e selecione 5 para autenticação segura versão 5. O valor padrão é 2.

Habilitar o modo agressivo na autenticação segura

As seleções válidas são Enabled (marcada) e Disabled (desmarcada). O valor padrão é Disabled (desmarcado).

Quando a seleção está Desabilitada (desmarcada), o controlador desabilita o modo agressivo DNP3 no subsistema de autenticação segura.

Quando a seleção está Habilitada (marcada), o controlador habilita o modo agressivo DNP3 no subsistema de autenticação segura.

Código de função crítico na autenticação segura

Critical Function Codes:

0	Non-Critical	8	Non-Critical	21	Critical	29	Critical
1	Non-Critical	9	Non-Critical	22	Non-Critical	30	Non-Critical
2	Critical	10	Non-Critical	23	Non-Critical	31	Critical
3	Critical	11	Non-Critical	24	Critical	129	Non-Critical
4	Critical	12	Non-Critical	25	Non-Critical	130	Non-Critical
5	Critical	13	Critical	26	Non-Critical		
6	Critical	14	Critical	27	Non-Critical		
7	Non-Critical	20	Critical	28	Non-Critical		

Este código de função crítico na página de configuração do escravo DNP3 é usado para definir a lista dos códigos de função críticos na autenticação segura. Um código de função crítico deve ser definido clicando no ícone de número para alterá-lo entre Crítico e Não crítico.

A tabela a seguir mostra o estado padrão dos códigos de função definidos no software Connected Components Workbench.

Códigos de função

Código de função	CFs críticos	Código de função	CFs críticos
0 (0x00)	não crítico	20 (0x14)	crítico
1 (0x01)	não crítico	21 (0x15)	crítico
2 (0x02)	crítico	22 (0x16)	não crítico
3 (0x04)	crítico	23 (0x17)	não crítico
4 (0x04)	crítico	24 (0x18)	crítico
5 (0x05)	crítico	25 (0x19)	não crítico
6 (0x06)	crítico	26 (0x1A)	não crítico
7 (0x07)	não crítico	27 (0x1B)	não crítico
8 (0x08)	não crítico	28 (0x1C)	não crítico
9 (0x09)	não crítico	29 (0x1D)	crítico
10 (0x0A)	não crítico	30 (0x1E)	não crítico
11 (0x0B)	não crítico	31 (0x1F)	crítico
12 (0x0C)	não crítico	129 (0x81)	não crítico
13 (0x0D)	crítico	130 (0x82)	não crítico
14 (0x0E)	crítico		

Intervalo de alteração da chave da sessão esperada (x1 min) na autenticação segura

Este parâmetro é usado para configurar o intervalo de alteração da chave da sessão esperada em minutos.

A faixa válida é de 1 a 120 mín. O valor padrão é de 15 min.

Quando o DNP3 mestre não altera a chave da sessão dentro do tempo configurado, o controlador invalida a chave da sessão e seu estado para cada usuário.

Contagem de alterações da chave da sessão esperada (x1 min) na autenticação segura

Este parâmetro é usado para configurar a contagem de alterações da chave da sessão esperada em minutos.

A faixa válida é de 1 a 10000. O valor padrão é 1000.

Tempo limite de resposta (x100 ms) na autenticação segura

Este parâmetro é usado para configurar o tempo limite de resposta em 100 ms.

A faixa válida é de 0 a 1200 (120 s). O valor padrão é 20 (2 s).

Contagem máxima de erros na autenticação segura

Este parâmetro é usado para configurar a contagem máxima de erros.

A faixa válida é de 0 a 10. O valor padrão é 2.

Algoritmo HMAC em autenticação segura

Este parâmetro é usado para configurar o algoritmo HMAC.

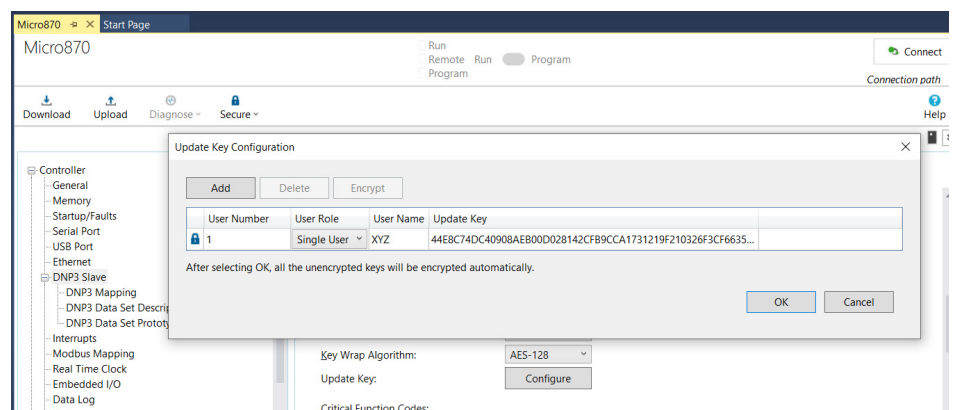
- SHA-1
 - Truncado para 4 octetos (serial)
 - Truncado para 10 octetos (em rede)
- SHA-256
 - Truncado para 8 octetos (serial)
 - Truncado para 16 octetos (em rede)

O valor padrão é SHA-256.

Atualizar chave na autenticação segura

Este parâmetro é usado para definir a autenticação segura das informações do usuário.

No software Connected Components Workbench, é possível criar as chaves de usuário na página de configuração do escravo DNP3.



Número do usuário

A faixa válida é de 1 a 65535.

Função do usuário

Uma lista suspensa com a seleção de várias funções que você pode definir para cada número de usuário (Visualizador, Operador, Engenheiro, Instalador, SECADM, SECAUD, RBACMNT e usuário único).

Nome do usuário

Defina o nome exclusivo para cada usuário, com até 32 caracteres (números, letras e símbolos).

Chave de atualização

A tecla a ser usada por cada usuário, com até 32 dígitos hexadecimais.

Siga estas etapas para criar um novo usuário.

1. Clique em Configure para abrir Update Key.
2. Clique em Add.
3. Insira o número do usuário, selecione a função do usuário, insira o nome do usuário e insira a chave de atualização.
4. Clique em Encrypt ou OK para criar o novo usuário.
5. Baixe o projeto para o controlador para atualizar as informações do usuário no controlador.

Para remover um usuário existente, siga estas etapas.

1. Clique em Configure para abrir Update Key.
2. Selecione o usuário a ser removido e clique em Delete.
3. Clique em OK para fechar a janela.
4. Baixe o projeto para o controlador para atualizar as informações do usuário no controlador.

Atualizar o método de alteração da chave e a chave da autoridade de certificação

IMPORTANTE Esse recurso está disponível somente na autenticação segura versão 5.

Uma chave de autoridade de certificação é uma chave simétrica (encriptada) ou pública (não criptografada) armazenada no controlador para autenticação com o mestre DNP3 quando uma solicitação de mudança de chave é processada.

Chave simétrica

The screenshot shows a dialog box titled 'Certificate Authority Key Configuration'. Inside, there is a section for 'Symmetric Key' with a large empty text box for input. Below the text box, a message states: 'After selecting OK, the Symmetric Key will be encrypted automatically.' At the bottom right of the dialog are 'OK' and 'Cancel' buttons. Below the dialog box, there are three configuration options: 'Update Key Change Method:' with a dropdown menu showing 'AES-256 / SHA-256-HMAC', 'Update Key:' with a 'Configure' button, and 'Certificate Authority Key:' with a 'Configure' button.

Chave pública

The screenshot shows a dialog box titled 'Certificate Authority Key Configuration'. Inside, there is a section for 'Public Key' with a large empty text box for input. At the bottom right of the dialog are 'OK' and 'Cancel' buttons. Below the dialog box, there are three configuration options: 'Update Key Change Method:' with a dropdown menu showing 'RSA-2048 / RSA SHA-256 / SHA-256-HMAC', 'Update Key:' with a 'Configure' button, and 'Certificate Authority Key:' with a 'Configure' button.

O tipo de chave usado no certificado é baseado na configuração Update Key Change Method que você selecionou na configuração. Para definir a chave, selecione uma das configurações a seguir.

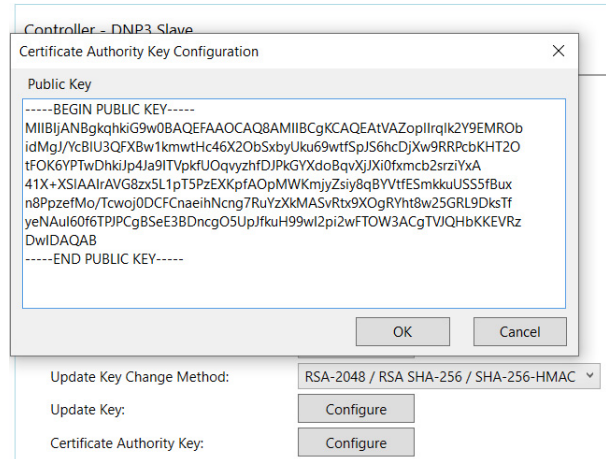
- Para usar a chave simétrica na autorização:
 - AES-128/SHA-1-HMAC
 - AES-256/SHA-256-HMAC
- Para usar a chave pública na autorização:
 - RSA-2048/RSA SHA-256/SHA-256/HMAC
 - RSA-3072/RSA SHA-256/SHA-256-HMAC

Para configurar a chave de autorização de certificado, clique em Configurar.

Para definir uma chave simétrica, insira 32 ou 64 caracteres no campo, dependendo do método de alteração da chave de atualização que você selecionou. Clique em OK para aceitar e criptografar a chave.

Para definir uma chave pública, você deve gerar uma chave pública RSA-2048 ou RSA-3072, dependendo do método de alteração da chave de atualização selecionado e inseri-la no campo. Clique em OK para aceitar as alterações. As chaves públicas não são criptografadas.

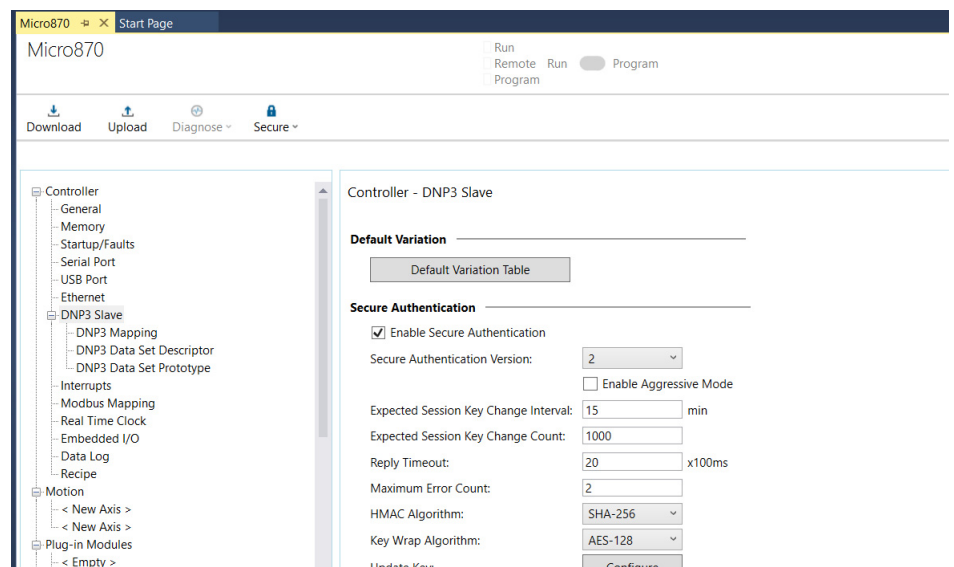
Exemplo de uma chave



Configuração de variação padrão

Esta configuração é usada para definir as variações padrão em uma resposta a uma solicitação coleta de Classe o.

No software Connected Components Workbench versão 20.01.00 ou posterior, você pode selecionar Default Variation Table na página de configuração do escravo DNP3 para alterar a configuração.



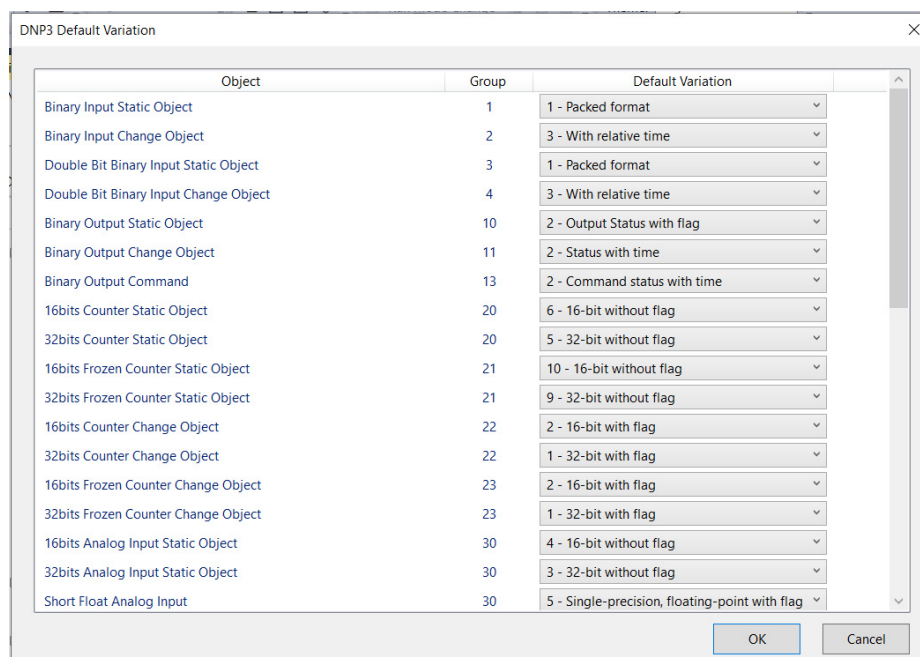


Tabela 7 mostra a estrutura do arquivo de configuração de variação padrão DNP3.

Tabela 7 - Arquivo de configuração de variação padrão DNP3

Grupo	Variação padrão para os seguintes objetos	Variação padrão	Variações padrão alternativas
1	Objeto estático de entrada binária	1 - Formato empacotado	0 - Todas as variações 2 - Com flag
2	Objeto de alteração de entrada binária	3 - Com tempo relativo	0 - Todas as variações 1 - Sem tempo 2 - Com tempo absoluto
3	Objeto estático de entrada binária de bit duplo	1 - Formato empacotado	0 - Todas as variações 2 - Com flag
4	Objeto de alteração de entrada binária de bit duplo	3 - Com tempo relativo	0 - Todas as variações 1 - Sem tempo 2 - Com tempo absoluto
10	Objeto estático de saída binária	2 - Status de saída com flag	0 - Todas as variações 1 - Formato empacotado
11	Objeto de alteração de saída binária	2 - Status com tempo	0 - Todas as variações 1 - Status sem tempo
13	Comando de saída binária	2 - Status de comando com tempo	0 - Todas as variações 1 - Status de comando sem tempo
20	Objeto estático do contador de 16 bits	6 - 16 bits sem flag	0 - Todas as variações 1 - 32 bits com flag 2 - 16 bits com flag 5 - 32 bits sem flag
	Objeto estático do contador de 32 bits	5 - 32 bits sem flag	0 - Todas as variações 1 - 32 bits com flag 2 - 16 bits com flag 6 - 16 bits sem flag
21	Objeto estático do contador de 16 bits congelado	10 - 16 bits sem flag	0 - Todas as variações 1 - 32 bits com flag 2 - 16 bits com flag 5 - 32 bits com flag e tempo 6 - 16 bits com flag e tempo 9 - 32 bits sem flag
	Objeto estático do contador de 32 bits congelado	9 - 32 bits sem flag	0 - Todas as variações 1 - 32 bits com flag 2 - 16 bits com flag 5 - 32 bits com flag e tempo 6 - 16 bits com flag e tempo 10 - 16 bits sem flag

Tabela 7 - Arquivo de configuração de variação padrão DNP3 (Continuação)

Grupo	Variação padrão para os seguintes objetos	Variação padrão	Variações padrão alternativas
22	Objeto de alteração do contador de 16 bits	2 - 16 bits com flag	0 - Todas as variações 1 - 32 bits com flag 5 - 32 bits com flag e tempo 6 - 16 bits com flag e tempo
	Objeto de alteração do contador de 32 bits	1 - 32 bits com flag	0 - Todas as variações 2 - 16 bits com flag 5 - 32 bits com flag e tempo 6 - 16 bits com flag e tempo
23	Objeto de alteração do contador de 16 bits congelado	2 - 16 bits com flag	0 - Todas as variações 1 - 32 bits com flag 5 - 32 bits com flag e tempo 6 - 16 bits com flag e tempo
	Objeto de alteração do contador de 32 bits congelado	1 - 32 bits com flag	0 - Todas as variações 2 - 16 bits com flag 5 - 32 bits com flag e tempo 6 - 16 bits com flag e tempo
30	Objeto estático de entrada analógica de 16 bits	4 - 16 bits sem flag	0 - Todas as variações 1 - 32 bits com flag 2 - 16 bits com flag 3 - 32 bits sem flag 5 - Precisão única, ponto flutuante com flag 5 - Precisão dupla, ponto flutuante com flag
	Objeto estático de entrada analógica de 32 bits	3 - 32 bits sem flag	0 - Todas as variações 1 - 32 bits com flag 2 - 16 bits com flag 4 - 16 bits sem flag 5 - Precisão única, ponto flutuante com flag 5 - Precisão dupla, ponto flutuante com flag
	Entrada analógica flutuante curta	5 - Precisão única, ponto flutuante com flag	0 - Todas as variações 1 - 32 bits com flag 2 - 16 bits com flag 3 - 32 bits sem flag 4 - 16 bits sem flag 5 - Precisão dupla, ponto flutuante com flag
32	Objeto de alteração da entrada analógica de 16 bits	2 - 16 bits sem tempo	0 - Todas as variações 1 - 32 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão única, ponto flutuante sem tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
	Objeto de alteração da entrada analógica de 32 bits	1 - 32 bits sem tempo	0 - Todas as variações 2 - 16 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão única, ponto flutuante sem tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
	Objeto de alteração de entrada analógica flutuante curta	5 - Precisão única, ponto flutuante sem tempo	0 - Todas as variações 1 - 32 bits sem tempo 2 - 16 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo

Tabela 7 - Arquivo de configuração de variação padrão DNP3 (Continuação)

Grupo	Variação padrão para os seguintes objetos	Variação padrão	Variações padrão alternativas
33	Objeto de alteração da entrada analógica de 16 bits congelada	2 - 16 bits sem tempo	0 - Todas as variações 1 - 32 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão única, ponto flutuante sem tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
	Objeto de alteração da entrada analógica de 32 bits congelada	1 - 32 bits sem tempo	0 - Todas as variações 2 - 16 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão única, ponto flutuante sem tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
	Objeto de alteração de entrada analógica de flutuação curta congelada	5 - Precisão única, ponto flutuante sem tempo	0 - Todas as variações 1 - 32 bits sem tempo 2 - 16 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
34	Banda morta de relatório de entrada analógica de 16 bits	1 - 16 bits	0 - Todas as variações 2 - 32 bits 5 - Precisão única, ponto flutuante
	Banda morta de relatório de entrada analógica de 32 bits	2 - 32 bits	0 - Todas as variações 1 - 16 bits 5 - Precisão única, ponto flutuante
	Banda morta de relatório de entrada analógica de flutuação curta	5 - Precisão única, ponto flutuante	0 - Todas as variações 1 - 16 bits 2 - 32 bits
40	Objeto estático de saída analógica de 16 bits	2 - 16 bits com flag	0 - Todas as variações 1 - 32 bits com flag 3 - Precisão única, ponto flutuante com flag 5 - Precisão dupla, ponto flutuante com flag
	Objeto estático de saída analógica de 32 bits	1 - 32 bits com flag	0 - Todas as variações 2 - 16 bits com flag 3 - Precisão única, ponto flutuante com flag 5 - Precisão dupla, ponto flutuante com flag
	Status da saída analógica flutuante curta	3 - Precisão única, ponto flutuante com flag	0 - Todas as variações 1 - 32 bits com flag 2 - 16 bits com flag 5 - Precisão dupla, ponto flutuante com flag
42	Objeto de alteração de saída analógica de 16 bits	2 - 16 bits sem tempo	0 - Todas as variações 1 - 32 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão única, ponto flutuante sem tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
	Objeto de alteração de saída analógica de 32 bits	1 - 32 bits sem tempo	0 - Todas as variações 2 - 16 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão única, ponto flutuante sem tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
	Objeto de alteração de saída analógica flutuante curta	5 - Precisão única, ponto flutuante sem tempo	0 - Todas as variações 1 - 32 bits sem tempo 2 - 16 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo

Tabela 7 - Arquivo de configuração de variação padrão DNP3 (Continuação)

Grupo	Variação padrão para os seguintes objetos	Variação padrão	Variações padrão alternativas
43	Objeto de alteração de comando de saída analógica de 16 bits	2 - 16 bits sem tempo	0 - Todas as variações 1 - 32 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão única, ponto flutuante sem tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
	Objeto de alteração de comando de saída analógica de 32 bits	1 - 32 bits sem tempo	0 - Todas as variações 2 - 16 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão única, ponto flutuante sem tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo
	Objeto de alteração de comando de saída analógica flutuante curta	5 - Precisão única, ponto flutuante sem tempo	0 - Todas as variações 1 - 32 bits sem tempo 2 - 16 bits sem tempo 3 - 32 bits com tempo 4 - 16 bits com tempo 5 - Precisão dupla, ponto flutuante sem tempo 5 - Precisão única, ponto flutuante com tempo 8 - Precisão dupla, ponto flutuante com tempo

Camada de aplicação escrava DNP3

Esta seção aborda os códigos de função da camada de aplicação escrava DNP3 e indicações internas. Todos os códigos de função suportados no controlador estão resumidos em [Códigos de função para DNP3 em controladores Micro870 na página 130](#).

Para obter detalhes sobre formatos de pacote para a solicitação e a resposta, consulte as especificações do protocolo DNP3.

Códigos de função

CONFIRM (Byte FC = 0x00)

00 – Confirmar

Um mestre DNP3 envia uma mensagem com este código de função para confirmar o recebimento de um fragmento de resposta. Em um ambiente geral, o controlador recebe uma resposta com este código de função. Mas o controlador pode gerar uma resposta com este código de função quando um mestre DNP3 envia uma solicitação com o bit CON definido no cabeçalho de controle da aplicação.

READ (FC Byte = 0x01)

01 – Ler

O código da função READ é usado por um mestre DNP3 para solicitar dados do controlador.

WRITE (FC Byte = 0x02)

02 – Gravar

O código da função WRITE é usado para gravar o conteúdo de objetos DNP3 do mestre DNP3 para o controlador. Este código de função é usado para limpar o bit IIN1.7 [DEVICE_RESTART], definir o tempo no controlador e baixar programas do usuário para o controlador.

SELECT (FC Byte = 0x03)

03 – Selecionar

O código de função SELECT é usado em conjunto com o código de função OPERATE como parte do método select-before-operate para emitir solicitações de controle. Este procedimento é usado para controlar objetos de saída binária (CROB) ou de saída analógica (AOB).

OPERATE (FC Byte = 0x04)

04 – Operar

Consulte [SELECT \(FC Byte = 0x03\) na página 104](#).

DIRECT_OPERATE (FC Byte = 0x05)

05 – Operação direta

Esta função de operação direta é semelhante ao código de função FC_OPERATE exceto que não é necessário nenhum comando de seleção precedente.

DIRECT_OPERATE_NR (FC Byte = 0x06)

06 – Operação direta sem resp

Consulte [DIRECT_OPERATE \(FC Byte = 0x05\)](#). Nenhuma mensagem de resposta é retornada quando esta solicitação é emitida a partir de um mestre DNP3.

IMMED_FREEZE (FC Byte = 0x07)

07 – Congelamento imediato

Ao receber uma solicitação com esta função, o controlador copia o valor atual de um ponto do contador para um local de memória separado associado ao mesmo ponto. O valor copiado permanece constante até a próxima operação de congelamento para o mesmo ponto.

IMMED_FREEZE_NR (FC Byte = 0x08)

08 – Congelamento imediato sem resposta

Consulte [IMMED_FREEZE \(FC Byte = 0x07\)](#). Nenhuma mensagem de resposta é retornada quando esta solicitação é emitida a partir de um mestre DNP3.

FREEZE_CLEAR (FC Byte = 0x09)

09 – Congelar e limpar

Ao receber uma solicitação com esta função, o controlador copia o valor atual para o valor congelado, em seguida atribui o imediatamente para o valor atual.

FREEZE_CLEAR_NR (FC Byte = 0x0A)

10 – Congelar e limpar sem resp

Consulte [FREEZE_CLEAR \(FC Byte = 0x09\)](#). Nenhuma mensagem de resposta é retornada quando esta solicitação é emitida a partir de um mestre DNP3.

COLD_RESTART (FC Byte = 0x0D)

13 – Partida a frio

Este código de função força o controlador a executar um reinício completo na energização.

WARM_RESTART (FC Byte = 0x0E)

14 – Reinício a quente

Este código de função força o controlador a executar um reset parcial.

INITIALIZE_APPL (FC Byte = 0x10)

16 – Inicializar aplicação

Este código de função é usado para inicializar o programa do usuário que foi baixado pelo software Connected Components Workbench.

START_APPL (FC Byte = 0x11)

17 – Iniciar a aplicação.

Este código de função é usado para iniciar o programa do usuário que foi baixado pelo software Connected Components Workbench.

STOP_APPL (FC Byte = 0x12)

18 – Parar a aplicação

Este código de função é usado para parar o programa do usuário que foi baixado pelo software Connected Components Workbench.

ENABLE_UNSOLICITED (FC Byte = 0x14)

20 – Habilitar mensagem não solicitada

Esta função é usada para habilitar dinamicamente mensagens não solicitadas geradas no controlador.

DISABLE_UNSOLICITED (FC Byte = 0x15)

21 – Desabilitar mensagem não solicitada

Esta função é usada para desabilitar dinamicamente mensagens não solicitadas geradas no controlador.

DELAY_MEASURE (FC Byte = 0x17)

23 – Medição de atraso, usada para procedimento não rede local

Este código de função é usado para medir o tempo de atraso do canal de comunicação.

RECORD_CURRENT_TIME (FC Byte = 0x18)

24 – Registrar hora atual, usada para procedimento de rede local

Este código de função é usado no procedimento para sincronizar os controladores que se comunicam através de uma rede local.

OPEN_FILE (FC Byte = 0x19)

25 – Abrir arquivo

Este código de função é usado para tornar um arquivo disponível para leitura ou gravação.

CLOSE_FILE (FC Byte = 0x1A)

26 – Fechar arquivo

Após a operação de leitura ou gravação do arquivo, este código de função é usado para desbloquear o arquivo.

DELETE_FILE (FC Byte = 0x1B)

27 – Excluir arquivo

Um mestre DNP3 usa este código de função para excluir um arquivo.

GET_FILE_INFO (FC Byte = 0x1C)

28 – Obter informações de arquivo

Este código de função é para o mestre recuperar informações sobre um arquivo no controlador.

AUTHENTICATE_FILE (FC Byte = 0x1D)

29 – Autenticar arquivo

Este código de função é usado para obter uma chave de autenticação que é necessária para abrir ou excluir um arquivo.

ABORT_FILE (FC Byte = 0x1E)

30 – Cancelar arquivo

Este código de função é usado para solicitar imediatamente o encerramento da operação de leitura/gravação atual e fechar o arquivo, sem salvar.

ACTIVATE_CONFIG (FC Byte = 0x1F)

31 – Ativar configuração

Este código de função é usado para começar a usar a configuração ou o código executável especificado pelos objetos incluídos na solicitação.

AUTHENTICATE_REQ (FC Byte = 0x20)

32 – Solicitação de autenticação

O mestre usa esse código de função ao enviar para o controlador mensagens de autenticação que exigem uma resposta

AUTHENTICATE_ERR (FC Byte = 0x21)

33 – Solicitação de autenticação sem resp

Este código de função é usado pelo mestre para enviar mensagens de autenticação quando não é necessária uma resposta de retorno.

RESPONSE (FC Byte = 0x22)

129 – Resposta

Todas as respostas usam este código de função, exceto as mensagens de resposta não solicitada.

UNSOLICITED_RESPONSE (FC Byte = 0x23)

130 – Resposta não solicitada

Respostas não solicitadas sempre usam este código de função independente de quais objetos DNP3 estão incluídos.

AUTHENTICATE_RESPONSE (FC Byte = 0x24)

131 – Resposta de autenticação

Este código de função é usado para emitir mensagens de autenticação para o mestre.

Indicações internas

Os bits de indicação interna são definidos sob as seguintes condições dos controladores:

- IIN1.0: ALL_STATIONS. Este bit é ativado quando é recebida uma mensagem de todas as estações.
- IIN1.1: CLASS_1_EVENTS. Este bit é definido quando estão disponíveis os dados de evento Classe 1.
- IIN1.2: CLASS_2_EVENTS. Este bit é definido quando estão disponíveis os dados de evento Classe 2.
- IIN1.3: CLASS_3_EVENTS. Este bit é definido quando estão disponíveis os dados de evento Classe 3.
- IIN1.4: NEED_TIME. Este bit é definido quando é necessária a sincronização de tempo.

- IIN1.5: LOCAL_CONTROL. Este bit é ativado quando o controlador está no modo não execução.
- IIN1.6: DEVICE_TROUBLE. Este bit é ativado quando o controlador está no modo com falha.
- IIN1.7: DEVICE_RESTART. Este bit é ativado quando o driver DNP3 é recém-configurado, na configuração do canal ou quando o controlador foi reiniciado. Para definir este bit durante a configuração do driver e a configuração do canal, é necessário selecionar Send Init. Não sol. Resp nula na configuração Reiniciar e definir o bit de status S:36/13 como 1 antes de fazer o download para o controlador.
- IIN2.0: NO_FUNC_CODE_SUPPORT. Este bit é ativado quando é recebida uma solicitação que tem um código de função desconhecido.
- IIN2.1: OBJECT_UNKNOWN. Este bit é ativado quando é recebida uma solicitação que tem um objeto desconhecido.
- IIN2.2: PARAMETER_ERROR. Este bit é definido quando é recebida uma solicitação com um campo de qualificador/intervalo que não pode ser processado.
- IIN2.3: EVENT_BUFFER_OVERFLOW. Este bit é ativado quando há uma condição de overflow do buffer de evento no controlador e pelo menos um evento não confirmado foi perdido.
- IIN2.4: ALREADY_EXECUTING. Não suportados.
- IIN2.5: CONFIG_CORRUPT. Este bit é definido quando são detectados um tipo de arquivo ruim e um número de arquivo ruim.
- IIN2.6: Reservado.
- IIN2.7: Reservado.

Objetos DNP3 e variáveis do controlador

Todos os objetos DNP3 suportados pelo controlador estão resumidos na [Tabela de implementação para controladores Micro870 na página 132](#).

As variáveis usadas nos objetos DNP3 não são as mesmas usadas no controlador, mas são similares. É necessário fazer o mapeamento entre variáveis em objetos DNP3 e variáveis do controlador.

Visão geral

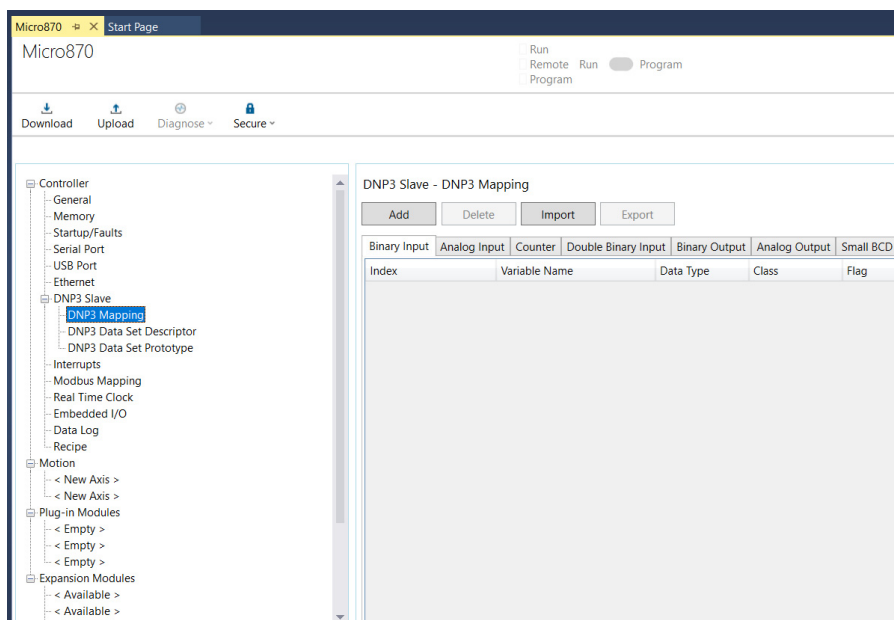
Os objetos de dados DNP3 que são implementados no controlador estão listados abaixo:

- Objeto de entrada binária DNP3
- Objeto de entrada binária de bit duplo DNP3
- Objeto de saída binária DNP3
- Objeto contador DNP3
- Objeto contador congelado DNP3
- Objeto de entrada analógica DNP3
- Objeto de saída analógica DNP3
- Objeto BCD DNP3
- Objeto Data-Set DNP3

Alguns dos objetos são divididos em vários arquivos de objeto para mapear dados no controlador.

- Objeto Counter — Objeto contador de 16 e 32 bits
- Objeto Analog Input — Objeto de entrada analógica de 16 e 32 bits e objeto de entrada analógica de ponto flutuante curto.
- Objeto Analog Output — Objeto de saída analógica de 16 e 32 bits e objeto de saída analógica de ponto flutuante curto.

Mapeamento DNP3 para controladores Micro870



Você pode criar os diferentes objetos de dados mapeando-os para as variáveis criadas no controlador. Você pode configurar o objeto Data para cada objeto DNP3 na página de configuração DNP3 Slave. As variáveis podem ser dos tipos de dados BOOL, INT, DINT ou REAL.

Dados do objeto DNP3

Tabela 8 - Relação entre o banco de dados do objeto DNP3 e as variáveis do Micro800

Objetos DNP			Variáveis Micro800	
Nome do objeto	Grupos relacionados	Índice máximo configurável	Nome do dado	Máximo de elementos configuráveis
Objeto de entrada binária	1, 2	4096	Objeto de entrada binária	256
Objeto de entrada binária de bit duplo	3, 4	2048	Objeto de entrada binária de bit duplo	256
Objeto de saída binária	10, 12	4096	Objeto de saída binária	256
Objeto contador	20, 22	256	Objeto contador de 16 bits	256
			Objeto contador de 32 bits	
Objeto contador congelado	21, 23	Reflexão do objeto contador que foi configurado	Reflexão do objeto contador de 16 bits	-
			Reflexão do objeto contador de 32 bits	
Objeto de entrada analógica	30, 32	256	Objeto de entrada analógica de 16 bits	256
			Objeto de entrada analógica de 32 bits	
			Objeto de entrada analógica de ponto flutuante curto	
Objeto de saída analógica	40, 41	256	Objeto de saída analógica de 16 bits	256
			Objeto de saída analógica de 32 bits	
			Objeto de saída analógica de ponto flutuante curto	

Tabela 8 - Relação entre o banco de dados do objeto DNP3 e as variáveis do Micro800 (Continuação)

Objetos DNP			Variáveis Micro800	
Nome do objeto	Grupos relacionados	Índice máximo configurável	Nome do dado	Máximo de elementos configuráveis
Objeto BCD	101	256	Objeto BCD pequeno	256
Objeto Data-Set	85, 87, 88	10	Objeto de protótipos de Data-Set	10
	86, 87, 88		Objeto de descritores de Data-Set	

Configuração do DNP3

Você pode configurar parâmetros de objeto para cada elemento, como nível de classe e informações de bit de flag. Essas informações são definidas durante a criação do objeto na janela de mapeamento de dados no mapeamento DNP3.

DNP3 Slave - DNP3 Mapping

Binary Input	Analog Input	Counter	Double Binary Input	Binary Output	Analog Output	Small BCD
Index	Variable Name	Data Type	Class	Flag	Mapped Points	
0	_JO_EM_DO_03	BOOL	0		1	

Objeto de conjunto de dados DNP3

Para criar no controlador um objeto de conjunto de dados a partir do subsistema DNP3, configure o objeto conjunto de dados de protótipos/descriptores no respectivo conjunto de dados do DNP3 escravo.

Micro870 - Start Page

Micro870

Controller

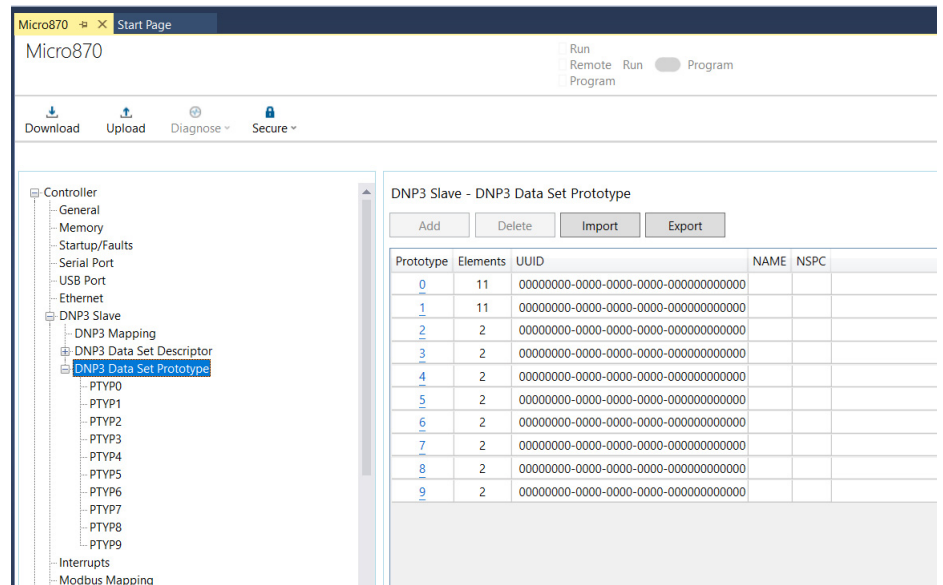
- General
- Memory
- Startup/Faults
- Serial Port
- USB Port
- Ethernet
- DNP3 Slave
 - DNP3 Mapping
 - DNP3 Data Set Descriptor**
 - DNP3 Data Set Prototype
- Interrupts
- Modbus Mapping
- Real Time Clock
- Embedded I/O
- Data Log
- Recipe
- Motion
 - < New Axis >
 - < New Axis >
- Plug-in Modules
 - < Empty >
 - < Empty >
 - < Empty >
- Expansion Modules
 - < Available >
 - < Available >

DNP3 Slave - DNP3 Data Set Descriptor

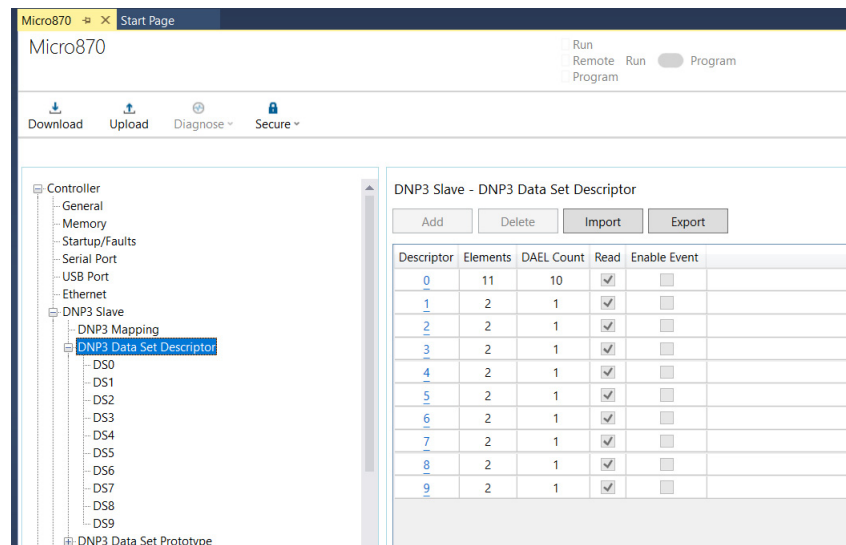
Descriptor	Elements	DAEL Count	Read	Enable Event
------------	----------	------------	------	--------------

Cada objeto de protótipos de conjunto de dados pode ter até 10 elementos de protótipos de conjunto de dados, e cada objeto de descritores de conjunto de dados pode ter até 10 elementos de descritores de conjunto de dados.

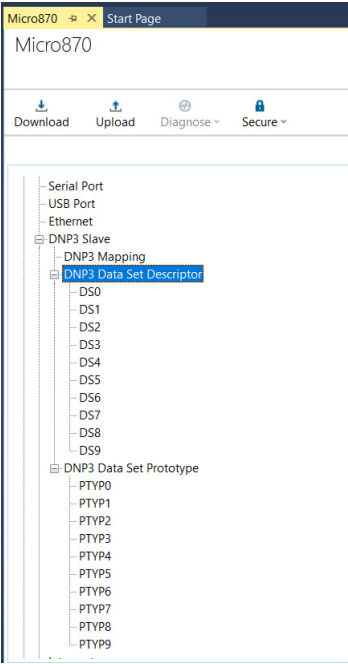
A título de exemplo, com a entrada protótipos de conjunto de dados, é possível criar qualquer número de objetos de protótipo de conjunto de dados na tela de configuração de DNP3 Data Set, até um máximo de 10 entradas.



A título de exemplo, com a entrada descritores de conjunto de dados, é possível criar qualquer número de objetos de descritor de conjunto de dados na tela de configuração de DNP3 Descriptor, até um máximo de 10 entradas.



Assim que os Protótipos e Descritores do Conjunto de Dados estiverem configurados, na página de configuração do escravo DNP3 do software Connected Components Workbench versão 20.01.00 ou posterior, você poderá ver o Descritor DNP3 DSX e o Protótipo PTYPX na respectiva ramificação DNP3 Data Set, onde X é os número de elementos de cada Protótipo ou Descritor.



Para DNP3 PTYPX, você pode configurar o controlador para construir os objetos de protótipo do conjunto de dados.

DNP3 Data Set Prototype - PTYP0

ⓘ

Index	Descriptor Code	DataType Code	Max Data Length (bytes)	Ancillary Value Length (bytes)	Ancillary Value
1	ID	NONE	0	1	0
2	UUID	NONE	0	16	00000000-0000-0000-0000-000000000000
3	DAEL	INT	2	0	
4	DAEL	VSTR	1	0	
5	DAEL	VSTR	1	0	
6	DAEL	VSTR	1	0	
7	DAEL	VSTR	1	0	
8	DAEL	VSTR	1	0	
9	DAEL	VSTR	1	0	
10	DAEL	VSTR	1	0	
11	DAEL	VSTR	1	0	

Para DNP3 DSX, você pode configurar o controlador para construir os objetos descritores do conjunto de dados.

DNP3 Data Set Descriptor - DS0

ⓘ

Index	Descriptor Code	DataType Code	Max Data Length (bytes)	Ancillary Value Length (bytes)	Ancillary Value	Point Address
1	ID	NONE	0	1	0	
2	DAEL	INT	2	0		BI / 4
3	DAEL	FLT	4	0		BI / 0
4	DAEL	OSTR	1	0		BI / 0
5	DAEL	VSTR	1	0		BI / 0
6	DAEL	VSTR	1	0		BI / 0
7	DAEL	VSTR	1	0		BI / 0
8	DAEL	VSTR	1	0		BI / 0
9	DAEL	VSTR	1	0		BI / 0
10	DAEL	VSTR	1	0		BI / 0
11	DAEL	VSTR	1	0		BI / 0

☒ Read ☒ Enable Event

Event Class: 3 Trigger Event: Disable Change Event:


Event Occurrence Condition:

Index	Point Address Type	Point Address
1	Empty	
2	Empty	
3	Empty	
4	Empty	

Parâmetros de configuração de protótipos do conjunto de dados

Esses parâmetros são usados para construir o objeto protótipos do conjunto de dados.

DNP3 Data Set Prototype - PTYP0



Index	Descriptor Code	Data Type Code	Max Data Length (bytes)	Ancillary Value Length (bytes)	Ancillary Value
1	ID	NONE	0	1	0
2	UUID	NONE	0	16	00000000-0000-0000-0000-000000000000
3	DAEL	VSTR	1	6	111111
4	DAEL	VSTR	1	0	
5	DAEL	VSTR	1	0	
6	DAEL	VSTR	1	0	
7	DAEL	VSTR	1	0	
8	DAEL	VSTR	1	0	
9	DAEL	VSTR	1	0	
10	DAEL	VSTR	1	0	
11	DAEL	VSTR	1	0	

Você pode adicionar ou excluir o índice em cada entrada do PTYPx.

Código do descritor: UUID para índice 2. NSPC/NAME/DAEL para índice 3 ou superior.

Código do tipo de dados: NONE para o índice 2. VSTR/UINT/INT/FLT/OSTR/BSTR/TIME para índice 3 ou superior.

Comprimento máximo de dados (bytes): 0 para o elemento 1. 0 a 255 para índice 3 ou superior.

Valor auxiliar: vetor binário em hexadecimal para o elemento 1. Grupos ASCII para índice 3 ou superior. Máximo de 32 bytes.

Parâmetros de configuração dos descritores do conjunto de dados

Esses parâmetros são usados para construir objetos protótipos do conjunto de dados.

☒ Read ☒ Enable Event

Event Class: Trigger Event: Disable Change Event:

Event Occurrence Condition:

Index	Point Address Type	Point Address
1	Empty	
2	Empty	
3	Empty	
4	Empty	

Read and Enable Event permite atribuir características ao descritor.

- Read – definida se o conjunto de dados puder ser lido.
- Enable Event – definido se o estado externo gerar um evento Data-Set.

Event Class – Com essa opção marcada, é exibida a classe de evento para permitir a atribuição de classe de evento a este descritor.

- 1 – Classe 1
- 2 – Classe 2
- 3 – Classe 3

Trigger Event: defina e ative este parâmetro para gerar um evento pela lógica ladder para gerar eventos cronometrados. Uma vez que este parâmetro esteja definido pela lógica ladder ou por comunicações, o controlador limpa-o automaticamente após gerar um evento no final da varredura.

Disable Change Event: defina e ative este parâmetro para suprimir os eventos gerados por qualquer condição de ocorrência de evento.

Event Occurrence Condition: podem ser configuradas as condições de evento de conjunto de dados para cada descritor de conjunto de dados pela condição de ocorrência de evento de conjunto de dados 1/2/3/4 no objeto de descritores de conjunto de dados DNP3. Quando um dos valores que está apontando para a condição de ocorrência de evento 1/2/3/4 é alterado ou os critérios são atendidos, o controlador gera um evento de conjunto de dados, recuperável usando o grupo de objetos 88, variação 1.

Event Occurrence Condition:

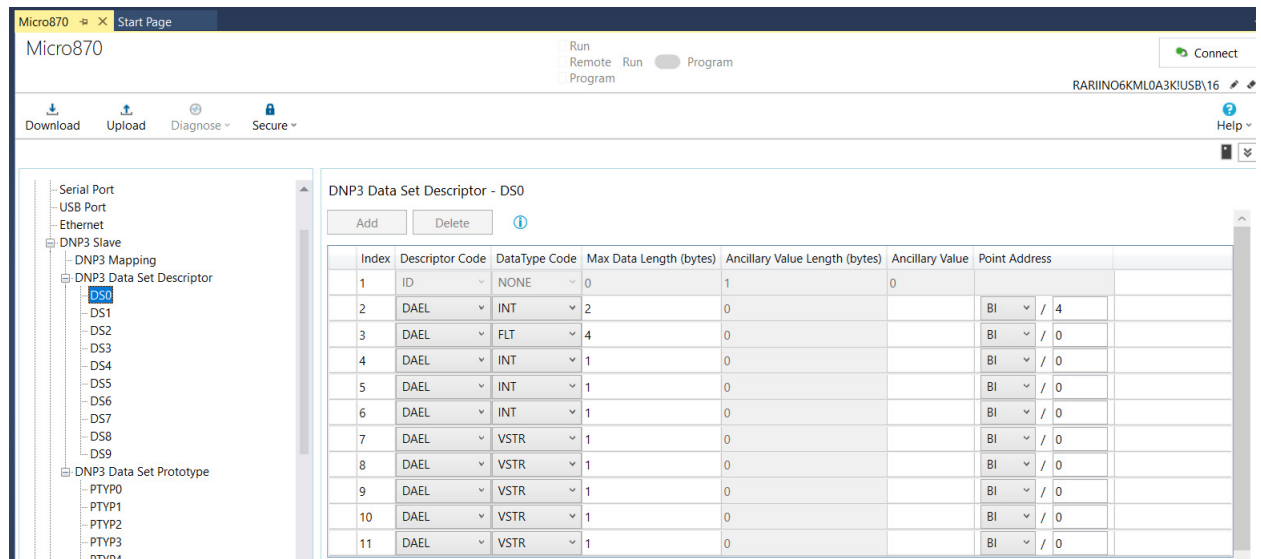
Index	Point Address Type	Point Address
1	Empty	
2	Empty	
3	Empty	
4	Empty	

A tabela a seguir mostra as condições suportadas para Point Addressing em Event Occurrence Conditions.

Tipo de endereço do ponto	Tipo de ponto	Índice do ponto	Condição de ocorrência de evento
Vazia	NONE: nenhum tipo de ponto está associado.	0	Nenhum evento é gerado.
Ponto DNP3 padrão	BI: entrada binária	0 a 4095	Quando o tipo de ponto e o índice do ponto estão apontando um ponto específico, se o valor do ponto for alterado, será gerado um evento.
	B2I: entrada de bit duplo	0 a 2047	
	CI: contador	0 a 511	
	AI: entrada analógica	0 a 767	
	BCD: ponto BCD	0 a 255	
Variável	Selecione o tag do controlador.	0	O evento é gerado com base na condição do tag de variável selecionado.

Observe que um evento de conjunto de dados pode consumir qualquer número de buffers de evento, dependendo da configuração do conjunto de dados. Isso só é aplicável para eventos de conjunto de dados. O evento para outros objetos consome um único buffer de evento. Ao usar eventos de conjunto de dados, aumente o número de eventos na configuração do escravo DNP3.

Configuração do elemento do descritor: cada elemento do descritor é configurado clicando no DSX individual no descritor do conjunto de dados DNP3. Adicione um novo índice clicando em Add em cada DSX.



Código de descrição: NAME, DAEL, PTYPE

Código do tipo de dados: VSTR, UINT, INT, FLT, OSTR, BSTR, TIME

Comprimento máximo de dados (bytes): 0 a 255

Valor auxiliar: qualquer grupo. Pode ser um vetor binário ou um grupo ASCII, de até 32 bytes.

Point Addressing em Descriptor Element Configuration: o valor do conjunto de dados para cada elemento do conjunto de dados é configurado por:

- Tipo de endereço do ponto
- Índice do ponto

Quando esses valores são configurados corretamente de acordo com os arquivos de dados suportados, o controlador responde com um objeto Group 87, Variation 1 preenchido com o valor no arquivo de dados. [Tabela 9 na página 116](#) mostra os arquivos de dados suportados para o endereçamento do ponto.

Tabela 9 – Tipo de endereço do ponto – Ponto DNP3 padrão

Tipo de endereço do ponto	Código do tipo de dados	Comprimento máximo de dados (bytes)	Tipo de ponto	Byte baixo de índice do ponto	Byte alto de índice do ponto
Padrão Ponto DNP3	NONE = 0	0	NONE = 0: Nenhum tipo de ponto está associado.	0	
	NONE = 0 UINT = 2 INT = 3 OSTR = 5 BSTR = 6 TIME = 7	0 0, 1, 2, ou 4 0, 1, 2, ou 4 0 a 255 0 a 255 0 a 6	BI = 1: Entrada binária	0 a 4.095 máx. Quando são usados os tipos de dados diferentes de OSTR e BSTR, o índice do ponto deve ser definido para um offset do ponto que seja divisível por 16.	
			B2I = 3: Entrada de bit duplo	0 a 2047 máx. Quando são usados os tipos de dados diferentes de OSTR e BSTR, o índice do ponto deve ser definido para um offset do ponto que seja divisível por 8.	
			CI = 20: Contador	0 a 511 máx.	
			AI = 30: Entrada analógica	0 a 767 máx.	
			BCD = 101: Ponto de BCD	0 a 255 máx.	

Quando o código do descritor é selecionado como PTYP, os parâmetros endereço do ponto para o elemento Descritor são substituídos por parâmetros de endereçamento de 10 pontos. Eles devem ser configurados na mesma ordem dos elementos DAEL nos Protótipos relevantes.

DNP3 Data Set Descriptor - DS0

Index	Descriptor Code	Data Type Code	Max Data Length (bytes)	Ancillary Value Length (bytes)	Ancillary Value	Point Address
1	ID	NONE	0	1	0	
2	PTYP	NONE	0	16	00000000-0000-0000-0000-000000000000	
	ID	NONE	0	1	0	
	UUID	NONE	0	16	00000000-0000-0000-0000-000000000000	
	DAEL	VSTR	1	6	111111	BI / 0
	DAEL	VSTR	1	0		BI / 0
	DAEL	VSTR	1	0		BI / 0
	DAEL	VSTR	1	0		BI / 0
	DAEL	VSTR	1	0		BI / 0
	DAEL	VSTR	1	0		BI / 0
	DAEL	VSTR	1	0		BI / 0
	DAEL	VSTR	1	0		BI / 0
	DAEL	VSTR	1	0		BI / 0
	DAEL	VSTR	1	0		BI / 0
4	DAEL	INT	1	0		BI / 0
5	DAEL	INT	1	0		BI / 0
6	DAEL	VSTR	1	0		BI / 0

Por exemplo, se Protótipo 0 incluir um Namespace no Índice 3 e Name no Índice 4, então o primeiro DAEL no Protótipo 0 estará no Índice 5.

DNP3 Data Set Prototype - PTYP0

Add
Delete

Index	Descriptor Code	Data Type Code	Max Data Length (bytes)	Ancillary Value Length (bytes)	Ancillary Value
1	ID	NONE	0	1	0
2	UUID	NONE	0	16	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
3	NSPC	NONE	0	16	Application Name
4	NAME	NONE	0	10	Fault Name
5	DAEL	UINT	2	27	Fault Code in System Status

DNP3 Data Set Descriptor - DSO

Add
Delete

Index	Descriptor Code	Data Type Code	Max Data Length (bytes)	Ancillary Value Length (bytes)	Ancillary Value	Point Address
1	ID	NONE	0	1	0	
2	PTYP	NONE	0	16	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	
	ID	NONE	0	1	0	
	UUID	NONE	0	16	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	
	NSPC	NONE	0	16	Application Name	
	NAME	NONE	0	10	Fault Name	
	DAEL	UINT	2	27	Fault Code in System Status	BI / 0
4	DAEL	INT	1	0		BI / 0
5	DAEL	INT	1	0		BI / 0
6	DAEL	VSTR	1	0		BI / 0

Flags de qualidade do objeto

O flag de objeto é composto de um grupo de 8 bits para alguns objetos DNP3. As tabelas abaixo mostram descrições de flag para cada objeto. Os flags ONLINE, RESTART, COMM_LOST, REMOTE_FORCED e LOCAL_FORCED são comuns a todos os tipos de grupos de objetos que contêm flags.

Há algumas regras para o flag de objeto ser definido ou limpo para cada bit pelo controlador. As regras abaixo também são aplicadas aos dados de eventos.

- Quando o controlador está no modo de não execução, o flag de objeto é sempre 0.
- Quando o controlador está no modo de execução e não existe um arquivo de configuração, é definido somente o flag on-line no flag do objeto.
- Quando o controlador está no modo de execução e existe um arquivo de configuração, os flags no flag do objeto são definidos de acordo com o byte superior dos arquivos de configuração.

Flags de objeto para entrada binária

Offset do bit	Nome	Descrição
0	On-line	0 quando o controlador está ou estava no modo não execução. 1 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador está ou estava no modo de execução e o arquivo de configuração existe.
1	REINICIAR	Sempre 0. Não utilizado.
2	COMM_LOST	Sempre 0. Não utilizado.
3	REMOTE_FORCED	Sempre 0. Não utilizado.
4	LOCAL_FORCED	Sempre 0. Não utilizado.
5	CHATTER_FILTER	Sempre 0. Não utilizado.
6	Reservado	Sempre 0. Não utilizado.
7	Estado	Reflete o estado do ponto de entrada binária.

Flags de objeto para entrada binária dupla

Offset do bit	Nome	Descrição
0	On-line	0 quando o controlador está ou estava no modo não execução. 1 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador está ou estava no modo de execução e o arquivo de configuração existe.
1	REINICIAR	Sempre 0. Não utilizado.
2	COMM_LOST	Sempre 0. Não utilizado.
3	REMOTE_FORCED	Sempre 0. Não utilizado.
4	LOCAL_FORCED	Sempre 0. Não utilizado.
5	CHATTER_FILTER	Sempre 0. Não utilizado.
6	Estado	Reflete o estado do ponto de entrada binária de bit duplo. LSB de bit duplo.
7	Estado	Reflete o estado do ponto de entrada binária de bit duplo. MSB de bit duplo.

Flags de objeto para saída binária

Offset do bit	Nome	Descrição
0	On-line	0 quando o controlador está ou estava no modo não execução. 1 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador estiver no modo de execução e o arquivo de configuração existir.
1	REINICIAR	Sempre 0. Não utilizado.
2	COMM_LOST	Sempre 0. Não utilizado.
3	REMOTE_FORCED	Sempre 0. Não utilizado.
4	LOCAL_FORCED	Sempre 0. Não utilizado.
5	Reservado	Sempre 0. Não utilizado.
6	Reservado	Sempre 0. Não utilizado.
7	Estado	Reflete o estado do ponto de saída binária.

Flags de objeto para contador

Offset do bit	Nome	Descrição
0	On-line	0 quando o controlador está ou estava no modo não execução. 1 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador estiver no modo de execução e o arquivo de configuração existir.
1	REINICIAR	0 quando o controlador está ou estava no modo não execução. 0 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador estiver no modo de execução e o arquivo de configuração existir.
2	COMM_LOST	
3	REMOTE_FORCED	
4	LOCAL_FORCED	
5	ROLLOVER	
6	DISCONTINUITY	
7	Reservado	

Flags de objeto para entrada analógica

Offset do bit	Nome	Descrição
0	On-line	0 quando o controlador está ou estava no modo não execução. 1 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador estiver no modo de execução e o arquivo de configuração existir.

Flags de objeto para entrada analógica (Continuação)

Offset do bit	Nome	Descrição
1	REINICIAR	0 quando o controlador está ou estava no modo não execução. 0 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador estiver no modo de execução e o arquivo de configuração existir.
2	COMM_LOST	
3	REMOTE_FORCED	
4	LOCAL_FORCED	
5	OVER_RANGE	
6	REFERENCE_ERR	
7	Reservado	

Flags de objeto para saída analógica

Offset do bit	Nome	Descrição
0	On-line	0 quando o controlador está ou estava no modo não execução. 1 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador estiver no modo de execução e o arquivo de configuração existir.
1	REINICIAR	0 quando o controlador está ou estava no modo não execução. 0 quando o controlador está ou estava no modo de execução e o arquivo de configuração não existe. Pode ser 1 quando o controlador estiver no modo de execução e o arquivo de configuração existir.
2	COMM_LOST	
3	REMOTE_FORCED	
4	LOCAL_FORCED	
5	Reservado	
6	Reservado	
7	Reservado	

Objeto de atributo de dispositivo DNP3

O objeto de atributo de dispositivo pode ser usado para identificar dispositivos escravos DNP3.

Com o controlador, algumas das variações são escritas para que você possa ler ou gravar seus próprios grupos em sua aplicação.

A propriedade R/W mostra se o objeto é Read Only, Read ou Write. Se a propriedade R/W for gravável, o valor que foi gravado pelo dispositivo mestre DNP3 é armazenado na memória não volátil.

O grupo de objetos do atributo do dispositivo é o. O intervalo suportado da variação é de 209 a 255.

Grupo de objetos 0, variações para o conjunto de atributos 0

Variação	Ler/gravar	Tipo de dados do atributo	Comprimento máximo em bytes	Descrição	Valor
209	somente leitura	UINT	1	Versão de autenticação segura suportada na estação de saída	5
210	somente leitura	UINT	1	Número de estatísticas de segurança disponíveis na estação de saída	18
212	somente leitura	UINT	1	Número de protótipos de conjunto de dados definidos pelo mestre	1
213	somente leitura	UINT	1	Número de protótipos de conjunto de dados definidos pela estação de trabalho	1
214	somente leitura	UINT	1	Número de conjuntos de dados definidos pelo mestre	1
215	somente leitura	UINT	1	Número de conjuntos de dados definidos pela estação de saída	1
216	somente leitura	UINT	1	Número máximo de saídas binárias por solicitação	1

Grupo de objetos 0, variações para o conjunto de atributos 0 (Continuação)

Variação	Ler/gravar	Tipo de dados do atributo	Comprimento máximo em bytes	Descrição	Valor
217	somente leitura	UINT	4	Precisão de temporização local	100 em microssegundos
218	somente leitura	UINT	4	Duração da precisão da temporização	100 em segundos
219	somente leitura	INT	1	Suporte para eventos de saída analógica	1
220	somente leitura	UINT	4	Índice máximo de saída analógica	0
221	somente leitura	UINT	4	Número de saídas analógicas	0
222	somente leitura	INT	1	Suporte para eventos de saída binária	1
223	somente leitura	UINT	4	Máximo Índice de saída binária	0
224	somente leitura	UINT	4	Número de saídas binárias	0
225	somente leitura	INT	1	Suporte para eventos de contador congelado	1
226	somente leitura	INT	1	Suporte para contadores congelados	1
227	somente leitura	INT	1	Suporte para eventos de contador	1
228	somente leitura	UINT	4	Índice máximo do contador	0
229	somente leitura	UINT	4	Número de pontos do contador	0
230	somente leitura	INT	1	Suporte para entradas analógicas congeladas	1
231	somente leitura	INT	1	Suporte para eventos de entrada analógica	1
232	somente leitura	UINT	4	Índice máximo de entrada analógica	0
233	somente leitura	UINT	4	Número de pontos de entrada analógica	0
234	somente leitura	INT	1	Suporte para eventos de entrada binária de bit duplo	1
235	somente leitura	UINT	4	Índice máximo de entrada binária de bit duplo	0
236	somente leitura	UINT	4	Número de pontos de entrada binários de bit duplo	0
237	somente leitura	INT	1	Suporte para eventos de entrada binária	1
238	somente leitura	UINT	4	Índice de entrada binária máx.	0
239	somente leitura	UINT	4	Número de pontos de entrada binária	0
240	somente leitura	UINT	4	Tamanho máximo do fragmento de transmissão	Tamanho da resposta
241	somente leitura	UINT	4	Tamanho máximo do fragmento de recebimento	Tamanho da resposta
242	somente leitura	VSTR	5	Versão do software do fabricante do dispositivo	Esta variação retorna o firmware FRN. FRN 1.00. Faixas suportadas: FRN x.yy, FRN x.yyy, FRN xx.yy, or FRN xx.yyy onde x, xxés 0 ~ 99 e yy, yyy 00 ~ 999. Por exemplo, FRN 1.00, FRN 1.05, FRN 12.05, FRN 102.27 ou FRN 103.117.
243	somente leitura	VSTR	6	Versão do hardware do fabricante do dispositivo	Faixas suportadas: HW SER x/REV yy onde x é A ~ F e yy é 00 ~ 31. Por exemplo, HW SER A/REV 01, HW SER B/REV 03 ou HW SER C/REV 31.
244	—	—	—	Reservado para atribuição futura	—

Grupo de objetos 0, variações para o conjunto de atributos 0 (Continuação)

Variação	Ler/gravar	Tipo de dados do atributo	Comprimento máximo em bytes	Descrição	Valor
245	Ler/gravar	VSTR	comprimento do valor da string, máx. de 255 bytes	Nome do local atribuído ao usuário	Padrão
246	Ler/gravar	VSTR	comprimento do valor da string, máx. de 255 bytes	Código/número de ID atribuído pelo usuário	Padrão
247	—	—	—	Reservado para atribuição futura	—
248	—	—	—	Reservado para atribuição futura	—
249	somente leitura	VSTR	6	Conformidade e subconjunto DNP	Esta variação retorna o nível do subconjunto e a versão do procedimento de teste. 3:2004.
250	somente leitura	VSTR	comprimento do valor da string	Nome e modelo do produto do fabricante do dispositivo	Faixas suportadas: 2080-L70E-24xxxN SER A onde xxx é QWB ou QBB. Por exemplo, 2080-L70E-QWBN SER A ou 2080-L70E-24QBBN SER A.
251	—	—	—	Reservado para atribuição futura	—
252	somente leitura	VSTR	19	Nome do fabricante do dispositivo	Esta variação retorna o nome da empresa, Rockwell Automation.
253	somente leitura	—	—	Reservado para atribuição futura	—
254	somente leitura	—	—	Solicitação de todos os atributos não específicos	Esta variável retorna todas as variações neste grupo exceto esta variação.
255	somente leitura	—	—	Lista de variações de atributos	Esta variação retorna a propriedade R/W para cada variação. 0 para somente leitura 1 para leitura ou gravação

Relatório de eventos

Esta seção aborda como gerar eventos DNP3 a partir de objetos de dados DNP3 e como relatar os eventos gerados por resposta com coleta ou por resposta não solicitada.

Gerar eventos

O controlador tem uma área de buffer separada que pode ser usada para registrar eventos DNP3 internamente.

O número máximo de eventos que podem ser registrados é 10.000 (consulte [Registro de eventos DNP3 10K na página 123](#)), independentemente do tipo de dados de evento. Nos controladores 2080-L70E24QxBN, um evento de Data-Set pode consumir vários números dos buffers de eventos.

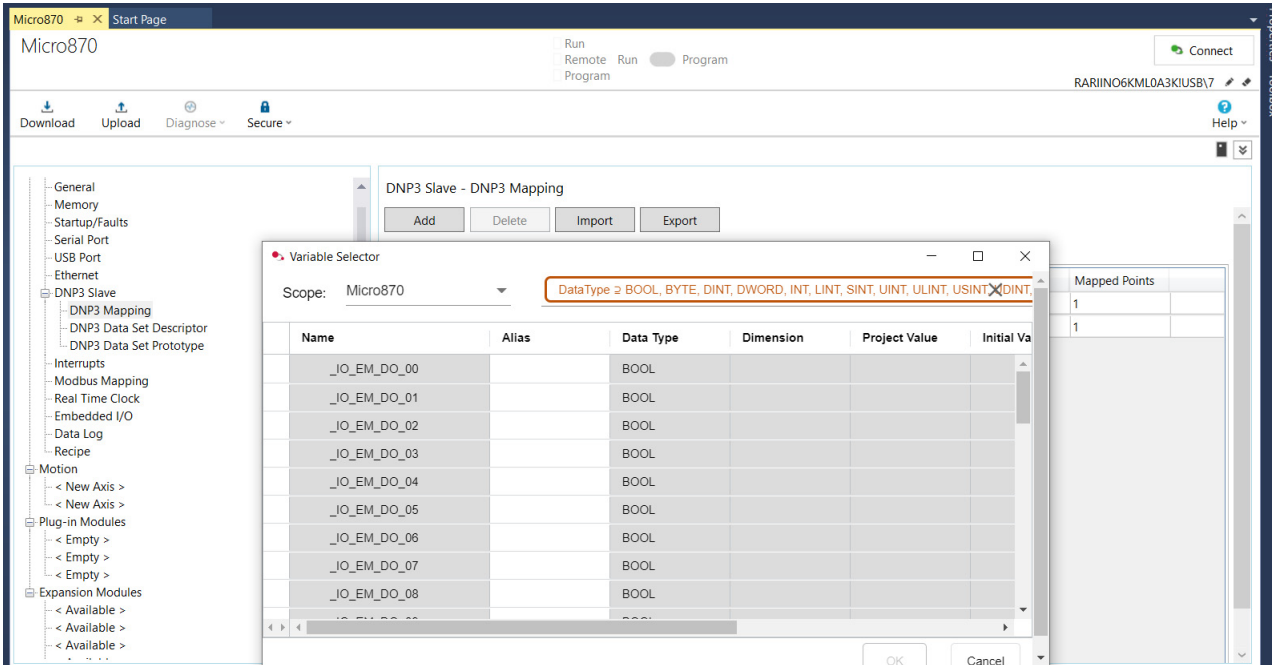
Se o número de eventos gerados atingir esse valor, o controlador definirá IIN2.3 [EVENT_BUFFER_OVERFLOW]. Eventos futuros não serão registrados até que os eventos registrados sejam reportados ao DNP3 mestre e o buffer esteja disponível novamente.

Os eventos registrados não são removidos até que sejam reportados com sucesso ao mestre DNP3. O evento registrado também pode ser removido quando ocorrer um dos eventos a seguir:

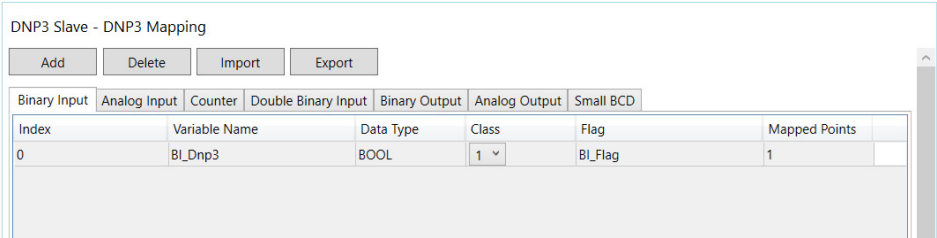
- Novo upgrade do firmware do sistema operacional
- Download de novo programa do usuário.

Figura 6 mostra como gerar eventos para um objeto Binary Input e um objeto Analog Input de 16 bits. Na configuração do escravo DNP3, os dados do objeto de entrada binária e os dados do objeto de entrada analógica de 16 bits são configurados na tabela de mapeamento do DNP3.

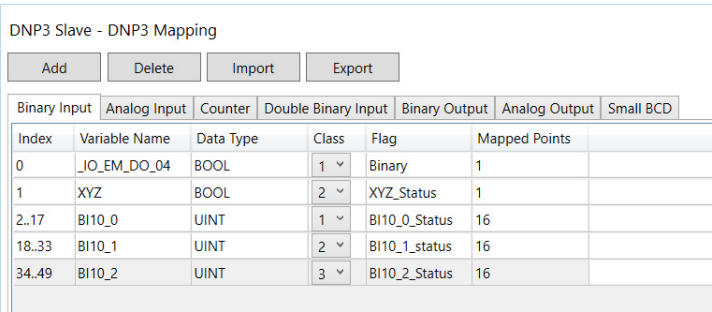
Figura 6 - Gerar eventos para objeto de entrada binária



Clique em Add para selecionar ou crie uma variável binária na tela do seletor de variáveis para o objeto Binary Input.



Existem dois métodos para criar o objeto de entrada binária. Você pode criar o objeto como um tipo de dados BOOL ou como um tipo de dados UINT. O tipo de dados UINT representa um vetor BOOL de 16 bits e esta representação é semelhante ao vetor BOOL usado nos controladores MicroLogix™ 1400.



A configuração na imagem acima mostra que os diferentes eventos de classe podem ser acionados quando a variável BOOL for acionada. Ao usar a variável UINT, qualquer um dos bits no UINT disparará o respectivo evento de classe.

Da mesma maneira, crie este objeto de entrada analógica de 16 bits na guia Analog Input e defina Index 0 como Class 1, Index 1 como Class 2 e Index 2 como Class 3.

DNP3 Slave - DNP3 Mapping

Add Delete Import Export

Index	Variable Name	Data Type	Class	Flag	Dead Band
0	AnalogValue1	REAL	1	AnalogV1	
1	AnalogValue2	REAL	2	AnalogV2	
2	AnalogValue3	REAL	3	AnalogV3	

Registro de eventos DNP3 10K

Podem ser registrados até 10.000 eventos. Para configurar o número de eventos, altere as respostas não solicitadas para a classe 1/2/3 para Enabled e defina 10.000 no campo Number of Events. Faça o download do projeto no controlador para que o ajuste de parâmetro entre em vigor.

Micro870

Remote Run Program

Connect

Download Upload Diagnose Secure

Controller - DNP3 Slave

Max Response Size: 2048 byte

Confirmation Timeout: 10000 ms

Select Timeout: 10 sec

☒ Enable Time Synchronization

Time Synchronization Interval: 0 min

Unsolicited Responses

Channel for Unsolicited Responses: Embedded Serial Port

☐ Restore Events After Power Cycle

☐ Send Initial Unsolicited Null Response on Restart

☐ Enable Unsolicited On Start Up

Number of Retries: 0

Unsolicited Responses for Class 1: Enabled Number of Events: 10000 Hold Time after Events: 5 sec

Unsolicited Responses for Class 2: Disabled Number of Events: 0

Unsolicited Responses for Class 3: Disabled Number of Events: 0

Sempre que houver uma mudança na configuração para o número de eventos, você deve fazer o download do projeto no controlador para que a configuração entre em vigor.

Controle de geração de eventos

O controlador verifica todos os elementos nos dados do objeto quanto a alterações no final de uma varredura e gera eventos sempre que necessário.

O método fundamental para ligar e desligar a geração de eventos por lógica ladder é atribuir a variável a ser usada para controlar ao campo Trigger Event e Disable Change Events.

O exemplo a seguir mostra como controlar a condição de geração de evento usando a banda morta para objetos de entrada analógica.

Neste exemplo, para o ponto 0 de entrada analógica de 16 bits, se o valor absoluto da diferença entre o valor atual da variável AnalogValue1 e o valor que foi enfileirado mais recentemente como um evento para esse ponto exceder o valor de banda morta da variável AnalogV1Deadband, então será gerado um evento para esse ponto.

DNP3 Slave - DNP3 Mapping

AddDeleteImportExport

Binary Input	Analog Input	Counter	Double Binary Input	Binary Output	Analog Output	Small BCD	
Index	Variable Name	Data Type	Class	Flag	Dead Band	Trigger Event	Disable Change Events
0	AnalogValue1	REAL	1	AnalogV1	AnalogV1Deadband	_JO_EM_DO_02	_JO_EM_DO_05
1	AnalogValue2	REAL	2	AnalogV2			
2	AnalogValue3	REAL	3	AnalogV3			

Evento de relatório por resposta com coleta

Quando um mestre DNP3 envia uma coleta para ler eventos de classe, todos os eventos registrados no buffer de eventos são relatados na resposta da coleta.

Defina o número de eventos para o respectivo evento de classe na página de configuração do escravo DNP3, conforme mostrado no exemplo a seguir.

Controller - DNP3 Slave

Max Response Size: 2048 byte

Confirmation Timeout: 10000 ms

Select Timeout: 10 sec

☒ Enable Time Synchronization

Time Synchronization Interval: 0 min

Unsolicited Responses

Channel for Unsolicited Responses: Embedded Serial Port

☐ Restore Events After Power Cycle

☐ Send Initial Unsolicited Null Response on Restart

☐ Enable Unsolicited On Start Up

Number of Retries: 0

Unsolicited Responses for Class 1: Disabled

Unsolicited Responses for Class 2: Disabled

Unsolicited Responses for Class 3: Disabled

Number of Events: 10

Number of Events: 10

Number of Events: 10

Evento de relatório por resposta não solicitada

Para iniciar e enviar respostas não solicitadas a um mestre DNP3, os parâmetros abaixo devem ser configurados corretamente. Para obter mais detalhes, consulte [Parâmetros de configuração da camada de aplicação escrava DNP3 na página 88](#).

- Nó mestre o
- Canal para resposta não solicitada
- Habilitar não solicitado na inicialização
- Habilitar não solicitado para Classe 1
- Habilitar não solicitado para Classe 2
- Habilitar não solicitado para Classe 3
- Na inicialização, enviar uma não solicitada inicial
- Número de eventos Classe 1
- Tempo de retenção após eventos de Classe 1 (x 1s)
- Número de eventos Classe 2
- Tempo de retenção após eventos de Classe 2 (x 1s)
- Número de eventos Classe 3
- Tempo de retenção após eventos de Classe 3 (x 1s)
- Dados do objeto DNP3
- Configuração do Objeto DNP3
- Conteúdo do arquivo de configuração

Em alguns casos, o controlador pode não enviar uma resposta não solicitada mesmo que os parâmetros estejam configurados corretamente

- Normalmente, quando o parâmetro Enable Unsolicited On Start Up está marcado, o controlador inicia uma resposta não solicitada com o código de função ENABLE_UNSOLICITED(20), se houver eventos registrados no buffer de eventos. No entanto, quando é recebida uma solicitação com o código de função DISABLE_UNSOLICITED(21), não será enviada uma resposta não solicitada.
- Quando o parâmetro Enable Unsolicited On Start Up estiver desmarcado, o controlador não dispara a resposta não solicitada até que seja recebida do DNP3 mestre uma solicitação com o código de função ENABLE_UNSOLICITED(20).

[Figura 7](#) mostra como iniciar e enviar a resposta não solicitada. O nó mestre o na página de configuração do escravo DNP3 indica que a resposta não solicitada é relatada ao mestre com o endereço do nó 3.

Figura 7 - Iniciar e enviar resposta não solicitada

Controller - DNP3 Slave

Max Response Size: 2048 byte

Confirmation Timeout: 10000 ms

Select Timeout: 10 sec

☒ Enable Time Synchronization

Time Synchronization Interval: 0 min

Unsolicited Responses

Channel for Unsolicited Responses: Embedded Serial Port

☐ Restore Events After Power Cycle

☐ Send Initial Unsolicited Null Response on Restart

☐ Enable Unsolicited On Start Up

Number of Retries: 0

Unsolicited Responses for Class 1: Enabled	Number of Events: 10	Hold Time after Events: 5 sec
Unsolicited Responses for Class 2: Enabled	Number of Events: 10	Hold Time after Events: 5 sec
Unsolicited Responses for Class 3: Enabled	Number of Events: 10	Hold Time after Events: 5 sec

O parâmetro de resposta não solicitada na configuração do escravo DNP3 indica que a resposta não solicitada é relatada para todas as portas de comunicação (serial e Ethernet) definidas. Neste exemplo, a resposta não solicitada inicial é enviada na inicialização e todos os eventos de classe 1, 2 e 3 são relatados. Como os tempos de espera são configurados para 5 segundos, os eventos gerados são relatados após 5 segundos.

Prevenção de colisão

O controlador atualmente suporta o primeiro dos dois métodos listados abaixo para evitar colisões.

- Detecção de dados transmitidos (linha TX/RX na comunicação RS-485).
- Detecção de portadora fora da banda (DCD na comunicação RS-232C).

Quando o controlador está conectado à rede RS-485, ele monitora todos os dados no link. Se o controlador está se preparando para transmitir um pacote e encontra o link ocupado, ele aguarda um intervalo definido por Backoff_Time até que não esteja mais ocupado.

Backoff_Time = Atraso pré-transmissão (x1 ms) + Atraso aleatório máximo (x1 ms)

O Atraso pré-transmissão (x1 ms) no arquivo Link Layer Channel Configuration é um atraso fixo e o Atraso máximo aleatório (x1 ms) no arquivo Channel Configuration é um atraso aleatório máximo para o canal 0 e canal 2. Você deve especificar esses parâmetros para obter o mecanismo de prevenção de colisão.

Após o Backoff_Time, o controlador tenta novamente, indefinidamente, ou até um número máximo configurável de novas tentativas. Se um máximo for usado, o protocolo considera isso como uma falha de link.

Sincronização de Tempo

O valor de tempo no relógio em tempo real (RTC) do controlador (relógio em tempo real do firmware) ou módulo plug-in (2080-MEMBAK-RTC2 ou 2080-SDMEMRTC-SC) é atualizado a cada 1 segundo.

O subsistema DNP3 e o RTC (firmware ou módulo plug-in) são sincronizados nas condições a seguir:

- energizar
- uma solicitação de sincronização de tempo do DNP3 mestre.

Na energização, o subsistema DNP3 recebe o tempo do controlador. Para o controlador adquirir o tempo correto, deve ser usado e habilitado um RTC do módulo plug-in antes de desligar e ligar a alimentação para adquirir o tempo correto do controlador. O RTC do firmware sempre reverte para uma data e hora padrão após desligar e ligar a alimentação, portanto, não é capaz de fornecer um tempo preciso para o subsistema DNP3.

Se não for usado um módulo RTC plug-in, então deve ser habilitado o RTC do firmware. Para habilitar o RTC do firmware, selecione Enable firmware real time clock.

Controller - Real Time Clock

i Date and time are only available when connected to controller.

Battery: Not Available

Date: Not Available

Time: Not Available

Set Date/Time...

☒ Allow real time clock to be changed in run mode

☒ Enable firmware real time clock

Quando há uma solicitação de gravação para sincronização de tempo de um mestre DNP3, o tempo no RTC (firmware ou módulo plug-in) é sincronizado com o tempo do mestre DNP3.

Esta tabela mostra a precisão do módulo plug-in 2080-MEMBAK-RTC2. Configure o intervalo de sincronização de tempo na configuração do escravo Micro870 DNP3 de acordo com esta tabela, para que um mestre DNP3 possa enviar a solicitação de sincronização de tempo periodicamente para tempos mais precisos no controlador.

Precisão do 2080-MEMBAK-RTC2

Precisão do RTC ⁽¹⁾	Temperatura ambiente
±5 s/mês	25 °C (77 °F)
±9 s/mês	-20 a +65 °C (-4 a +149 °F)

(1) Esses números são valores máximos de pior caso para um mês de 31 dias.

Diagnóstico

Os erros em um subsistema escravo DNP3 são registrados na página Communication Diagnostics no software Connected Components Workbench.

Figura 8 - Diagnóstico de comunicação para porta serial incorporada

The screenshot shows the 'Micro870 - Communication Diagnostics' window. The 'Communication' dropdown is set to 'Serial Port' and the 'Channel' dropdown is set to 'Embedded Serial'. The 'Driver' is listed as 'DNP3 Slave'. A 'Reset Counters' button is visible. Below the settings, the 'Diagnostic Counters' section displays a table of communication statistics.

Diagnostic Counters			
Channel Status:	1	NAK Messages Received:	0
Messages Received:	0	Messages Sent:	0
Messages Retried:	0	Messages Undelivered:	0
Bytes Received on Channel:	0	Bytes Sent on Channel:	0
Frames Received on Channel:	0	Frames Sent on Channel:	0
Fragments Received on Channel:	0	Fragments Sent on Channel:	0
Link Layer Error Count:	0	Last Link Layer Error Code:	0
Application Layer Error Count:	0	Last Application Layer Error Code:	0
Transport Layer Error Count:	0	Last Transport Layer Error Code:	0
Physical Layer Errors:	0	Last Physical Layer Error Code:	0
Current Session Status:	0	Session Events Overflowed:	0
Session Events Confirmed:	0	Session Events Sent:	0

Figura 9 - Diagnóstico de comunicação para o módulo plug-in 2080-SERIALISOL

The screenshot shows the 'Micro870 - Communication Diagnostics' window. The 'Communication' dropdown is set to 'Serial Port' and the 'Channel' dropdown is set to 'Slot 2 | 2080-SERIALISOL at port 6'. The 'Driver' is listed as 'DNP3 Slave'. A 'Reset Counters' button is visible. Below the settings, the 'Diagnostic Counters' section displays a table of communication statistics.

Diagnostic Counters	
Channel Status:	NAK Messages Received:
Messages Received:	Messages Sent:
Messages Retried:	Messages Undelivered:
Bytes Received on Channel:	Bytes Sent on Channel:
Frames Received on Channel:	Frames Sent on Channel:
Fragments Received on Channel:	Fragments Sent on Channel:
Link Layer Error Count:	Last Link Layer Error Code:
Application Layer Error Count:	Last Application Layer Error Code:
Transport Layer Error Count:	Last Transport Layer Error Code:
Physical Layer Errors:	Last Physical Layer Error Code:
Current Session Status:	Session Events Overflowed:
Session Events Confirmed:	Session Events Sent:

Diagnóstico para canal Ethernet

Os contadores de diagnóstico e erros no subsistema escravo DNP3 para o canal Ethernet são mostrados no software Connected Components Workbench. Clique em Diagnostic para exibir a página de diagnóstico.

Figura 10 - Diagnóstico de comunicação para porta Ethernet

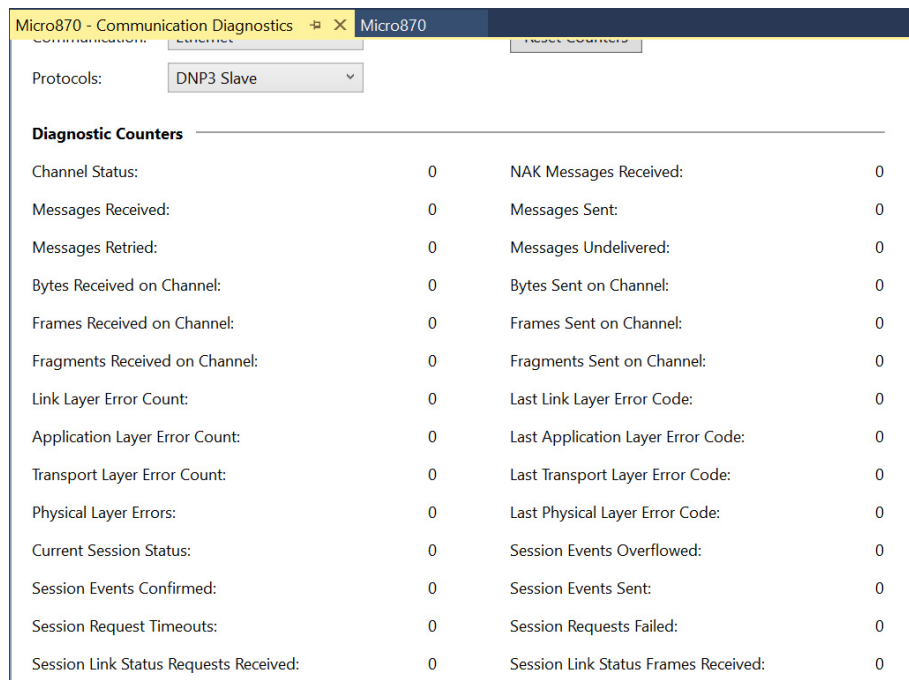


Tabela 10 - Códigos de erro

Valor (DEC)	Mnemônica	Descrição
0	ERROR_PHYS_TRANSMIT	Erro retornado da rotina de transmissão de destino
1	ERROR_PHYS_CHAR_TIMEOUT	Ocorreu o tempo limite intercaracteres
2	ERROR_PHYS_REMOTE_CLOSE	Lado remoto da conexão fechada do canal
3	ERROR_LINK_FRAME_LENGTH	O frame de entrada é muito curto ou excedeu o tamanho do buffer
4	ERROR_LINK_ADDRESS_DESCONHECIDO	O frame recebido era para um endereço de link desconhecido
5	ERROR_LINK_ILLEGAL_FUNCTION	Código de função de link ilegal no frame recebido
6	ERROR_LINK_INVALID_CHECKSUM	Checksum ou CRC inválido
7	ERROR_LINK_NOT_RESET	O link não foi reinicializado, frame rejeitado
8	ERROR_LINK_FCB	Recebido bit de contagem de frame inválido
9	ERROR_LINK_INVALID_START_CHAR	Não recebeu o caractere de sincronização inicial correto
10	ERROR_LINK_FRAME_TIMEOUT	O frame inteiro não foi recebido em um tempo especificado
11	ERROR_LINK_CNFM_TIMEOUT	A confirmação do link não foi recebida no tempo especificado
12	ERROR_LINK_STATUS_TIMEOUT	Resposta de status do link não recebida no tempo especificado
Os seguintes erros de link são usados por 101/103		
13	ERROR_LINK_WRONG_SESN	A resposta não era da sessão esperada
14	ERROR_LINK_WRONG_REPLY	Recebida resposta inesperada, frame rejeitado
15	ERROR_LINK_INVALID_2ND_CHAR	Não recebido o segundo caractere de sincronização correto
16	ERROR_LINK_INVALID_END_CHAR	Não recebido o caractere de sincronização final correto
17	ERROR_LINK_MISMATCHING_LENGTH	Os bytes de comprimento variável no frame FT1.2 não correspondem
18	ERROR_LINK_INV_DIR	Bit dir inválido recebido no octeto de controle
Os erros de link a seguir são usados por 104		
19	ERROR_LINK_NO_CNFM_RECEIVED	Confirmação de 104 U-formato APDU não recebida
20	ERROR_LINK_NO_ACK_RECEIVED	Reconhecido de 104 APDU no formato I não recebido

Tabela 10 - Códigos de erro (Continuação)

Valor (DEC)	Mnemônica	Descrição
21	ERROR_LINK_SEQUENCE_UNKNOWN	Número de sequência de confirmação desconhecido na APDU recebida
22	ERROR_LINK_OUT_OF_SEQUENCE	APDU recebida fora de sequência com a APDU anterior
Erros da camada de transporte		
23	ERROR_TPRT_SEQUENCE_ERROR	Erro de número de sequência

Diagnóstico para autenticação segura

As informações de diagnóstico para autenticação segura são mostradas no software Connected Components Workbench. Clique em Diagnostic para exibir a página de diagnóstico.

Micro870 - Communication Diagnostics			
Diagnostic Counters			
Channel Status:	1	NAK Messages Received:	0
Messages Received:	0	Messages Sent:	0
Messages Retried:	0	Messages Undelivered:	0
Bytes Received on Channel:	0	Bytes Sent on Channel:	0
Frames Received on Channel:	0	Frames Sent on Channel:	0
Fragments Received on Channel:	0	Fragments Sent on Channel:	0
Link Layer Error Count:	0	Last Link Layer Error Code:	0
Application Layer Error Count:	0	Last Application Layer Error Code:	0
Transport Layer Error Count:	0	Last Transport Layer Error Code:	0
Physical Layer Errors:	0	Last Physical Layer Error Code:	0
Current Session Status:	0	Session Events Overflowed:	0
Session Events Confirmed:	0	Session Events Sent:	0
Session Request Timeouts:	0	Session Requests Failed:	0
Session Link Status Requests Received:	0	Session Link Status Frames Received:	0
Session Secure Authentication Messages Received:	0	Session Secure Authentication Messages Sent:	0
Session Secure Authentication Response Timeouts:	0	Session Authentication Key Exchanges:	0

Códigos de função

Estas tabelas mostram os códigos de função da camada da aplicação implementados no controlador.

Tabela 11 - Códigos de função para DNP3 em controladores Micro870

Tipo de mensagem	Código de função	Nome	Suporte para Micro870	Descrição
Confirmation	0 (0x00)	CONFIRM	Sim	Análises/envios do controlador
Solicitação	1 (0x01)	READ	Sim	Análises do controlador
Solicitação	2 (0x02)	WRITE	Sim	Análises do controlador
Solicitação	3 (0x03)	SELECT	Sim	Análises do controlador
Solicitação	4 (0x04)	OPERATE	Sim	Análises do controlador
Solicitação	5 (0x05)	DIRECT_OPERATE	Sim	Análises do controlador
Solicitação	6 (0x06)	DIRECT_OPERATE_NR	Sim	Análises do controlador
Solicitação	7 (0x07)	IMMED_FREEZE	Sim	Análises do controlador
Solicitação	8 (0x08)	IMMED_FREEZE_NR	Sim	Análises do controlador
Solicitação	9 (0x09)	FREEZE_CLEAR	Sim	Análises do controlador
Solicitação	10 (0x0A)	FREEZE_CLEAR_NR	Sim	Análises do controlador

Tabela 11 - Códigos de função para DNP3 em controladores Micro870 (Continuação)

Tipo de mensagem	Código de função	Nome	Suporte para Micro870	Descrição
Solicitação	11 (0x0B)	FREEZE_AT_TIME	Não	
Solicitação	12 (0x0C)	FREEZE_AT_TIME_NR	Não	
Solicitação	13 (0x0D)	COLD_RESTART	Sim	Análises do controlador. O controlador não deve estar no modo de execução e nenhum programa e arquivo deve estar no estado aberto.
Solicitação	14 (0x0E)	WARM_RESTART	Não	Análises do controlador
Solicitação	15 (0x0F)	INITIALIZE_DATA	Não	Obsoleto
Solicitação	16 (0x10)	INITIALIZE_APPL	Sim	Análises do controlador. Remove a falha e altera o modo do controlador para Programa remoto.
Solicitação	17 (0x11)	START_APPL	Sim	Análises do controlador. Remove a falha e altera o modo do controlador para Operação remota.
Solicitação	18 (0x12)	STOP_APPL	Sim	Análises do controlador. Muda o modo do controlador para Programa remoto.
Solicitação	19 (0x13)	SAVE_CONFIG	Não	Descontinuado
Solicitação	20 (0x14)	ENABLE_UNSOLICITED	Sim	Análises do controlador
Solicitação	21 (0x15)	DISABLE_UNSOLICITED	Sim	Análises do controlador
Solicitação	22 (0x16)	ASSIGN_CLASS	Não	
Solicitação	23 (0x17)	DELAY_MEASURE	Sim	Análises do controlador. Usado para não rede local
Solicitação	24 (0x18)	RECORD_CURRENT_TIME	Não	Análises do controlador. Usado para rede local
Solicitação	25 (0x19)	OPEN_FILE	Sim	Análises do controlador
Solicitação	26 (0x1A)	CLOSE_FILE	Sim	Análises do controlador
Solicitação	27 (0x1B)	DELETE_FILE	Sim	Análises do controlador
Solicitação	28 (0x1C)	GET_FILE_INFO	Não	Análises do controlador
Solicitação	29 (0x1D)	AUTHENTICATE_FILE	Sim	Análises do controlador
Solicitação	30 (0x1E)	ABORT_FILE	Não	Análises do controlador
Solicitação	31 (0x1F)	ACTIVATE_CONFIG	Não	Análises do controlador
Solicitação	32 (0x20)	AUTHENTICATE_REQ	Não	Análises do controlador
Solicitação	33 (0x21)	AUTHENTICATE_ERR	Não	Análises do controlador
Resposta	34 (0x22)	Resposta	Sim	O controlador envia
Resposta	35 (0x23)	UNSOLICITED_RESPONSE	Sim	O controlador envia
Resposta	36 (0x24)	AUTHENTICATE_RESPONSE	Não	O controlador envia
	37 (0x25) a 47 (0x2F)		Não	Reservado

Tabela de implementação

Os controladores Micro870 (2080-L70E-24QxBN) suportam o subconjunto de certificação DNP3 nível 2.

[Tabela 12](#) identifica quais grupos de objetos e variações, códigos de função e qualificadores o dispositivo suporta em solicitações e respostas. As colunas Request and Response identificam todas as solicitações e respostas que podem ser enviadas/analísadas por um mestre DNP3 ou que devem ser analisadas/ enviadas pelo controlador.

A tabela de implementação lista todas as funcionalidades exigidas pelo DNP3 mestre ou pelo controlador, conforme definido nos Procedimentos de teste de conformidade IED DNP3. Qualquer funcionalidade além do nível de subconjunto mais alto suportado é indicada pelas células da tabela em cinza.

Tabela 12 – Tabela de implementação para controladores Micro870

Variação e grupo de objetos DNP			Solicitação DNP3 mestre pode emitir O controlador deve analisar	Resposta DNP3 mestre deve analisar O controlador pode emitir		
Núm. do grupo	Núm. Variável	Descrição	Códigos de função (dec)	Códigos de qualificação (hex)	Códigos de função (dec)	Códigos de qualificação (hex)
0	211 a 239 241 a 243 248 a 250 252	Atributo do dispositivo	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00 (partida/parada)
0	240 245 a 247	Atributo do dispositivo	1 (leitura) 2 (gravação)	00, 01 (partida/parada) 06 (sem faixa ou todos) 00, 01 (partida/parada)	129 (resposta)	00 (partida/parada)
0	254	Atributo de dispositivo – Solicitação não específica de todos os atributos	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)		
0	255	Atributos do dispositivo – Lista de variações de atributos	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00 (partida/parada)
1	0	Entrada binária – Qualquer variação	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)		
1	1	Entrada binária – Formato empacotado	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
1	2	Entrada binária – com flags	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
2	0	Evento de entrada binária – Qualquer variação	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
2	1	Evento de entrada binária – sem tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
2	2	Evento de entrada binária – com tempo absoluto	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
2	3	Evento de entrada binária – com tempo relativo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
3	0	Entrada binária de bit duplo – qualquer variação	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)		
3	1	Entrada binária de bit duplo – formato empacotado	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
3	2	Entrada binária de bit duplo – com flags	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
4	0	Evento de entrada binária de bit duplo – qualquer variação	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
4	1	Evento de entrada binária de bit duplo – sem tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
4	2	Evento de entrada binária de bit duplo – com tempo absoluto	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
4	3	Evento de entrada binária de bit duplo – com tempo relativo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
10	0	Saída binária – qualquer variação	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)		
10	2	Saída binária – status de saída com flags	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
12	1	Comando binário – bloco de saída a relé de controle (CROB)	3 (select) 4 (operar) 5 (op. direta) 6 (dir. op, no ack)	17, 28 (índice)	129 (resposta)	eco da solicitação
20	0	Contador – qualquer variação	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)		

Tabela 12 - Tabela de implementação para controladores Micro870 (Continuação)

Variação e grupo de objetos DNP			Solicitação DNP3 mestre pode emitir O controlador deve analisar		Resposta DNP3 mestre deve analisar O controlador pode emitir	
Núm. do grupo	Núm. Variável	Descrição	Códigos de função (dec)	Códigos de qualificação (hex)	Códigos de função (dec)	Códigos de qualificação (hex)
20	1	Contador - 32 bits com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
20	1	Contador - 32 bits com flag	7 (freeze) 8 (freeze noack) 9 (congelar limpo) 10 (frz. cl. noack)	06 (sem faixa ou todos)		
20	2	Contador - 16 bits com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
20	2	Contador - 16 bits com flag	7 (freeze) 8 (freeze noack) 9 (congelar limpo) 10 (frz. cl. noack)	06 (sem faixa ou todos)		
20	5	Contador - 32 bits sem flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
20	5	Contador - 32 bits sem flag	7 (freeze) 8 (freeze noack) 9 (congelar limpo) 10 (frz. cl. noack)	06 (sem faixa ou todos)		
20	6	Contador - 16 bits sem flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
20	6	Contador - 16 bits sem flag	7 (freeze) 8 (freeze noack) 9 (congelar limpo) 10 (frz. cl. noack)	06 (sem faixa ou todos)		
21	0	Contador congelado - qualquer variação	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)		
21	1	Contador congelado - 32 bits com flag	1 (leitura)	06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
21	2	Contador congelado - 16 bits com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
21	5	Contador congelado - 32 bits com flag e tempo	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
21	6	Contador congelado - 16 bits com flag e tempo	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
21	9	Contador congelado - 32 bits sem flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
21	10	Contador congelado - 16 bits sem flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
22	0	Evento do contador - qualquer variação	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
22	1	Evento do contador - 32 bits com flag	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
22	2	Evento do contador - 16 bits com flag	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
22	5	Evento do contador - 32 bits com flag e tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
22	6	Evento do contador - 16 bits com flag e tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
23	0	Evento do contador congelado - qualquer variação	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
23	1	Evento de contador congelado - 32 bits com flag	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)

Tabela 12 - Tabela de implementação para controladores Micro870 (Continuação)

Variação e grupo de objetos DNP			Solicitação DNP3 mestre pode emitir O controlador deve analisar		Resposta DNP3 mestre deve analisar O controlador pode emitir	
Núm. do grupo	Núm. Variável	Descrição	Códigos de função (dec)	Códigos de qualificação (hex)	Códigos de função (dec)	Códigos de qualificação (hex)
23	2	Evento de contador congelado - 16 bits com flag	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
23	5	Evento de contador congelado - 32 bits com flag e tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
23	6	Evento de contador congelado - 16 bits com flag e tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
30	0	Entrada analógica - qualquer variação	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)		
30	1	Entrada analógica - 32 bits com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
30	2	Entrada analógica - 16 bits com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
30	3	Entrada analógica - 32 bits sem flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
30	4	Entrada analógica - 16 bits sem flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
30	5	Entrada analógica - Single-prec flt-pt com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
32	0	Evento de entrada analógica - qualquer variação	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
32	1	Evento de entrada analógica - 32 bits sem tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
32	2	Evento de entrada analógica - 16 bits sem tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
32	3	Evento de entrada analógica - 32 bits com tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
32	4	Evento de entrada analógica - 16 bits com tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
32	5	Evento de entrada analógica - Single-prec flt-pt sem tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
32	7	Evento de entrada analógica - Single-prec flt-pt com tempo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	17, 28 (índice)
40	0	Status da saída analógica - qualquer variação	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)		
40	1	Status da saída analógica - 32 bits com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
40	2	Status da saída analógica - 16 bits com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
40	3	Status de saída analógica - Single-prec flt-pt com flag	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos)	129 (resposta)	00, 01 (partida/parada)
41	1	Saída analógica - 32 bits	3 (select) 4 (operar) 5 (op. direta) 6 (dir. op, no ack)	17, 27, 28 (índice)	129 (resposta)	eco da solicitação
41	2	Saída analógica - 16 bits	3 (select) 4 (operar) 5 (op. direta) 6 (dir. op, no ack)	17, 27, 28 (índice)	129 (resposta)	eco da solicitação

Tabela 12 - Tabela de implementação para controladores Micro870 (Continuação)

Variação e grupo de objetos DNP			Solicitação DNP3 mestre pode emitir O controlador deve analisar		Resposta DNP3 mestre deve analisar O controlador pode emitir	
Núm. do grupo	Núm. Variável	Descrição	Códigos de função (dec)	Códigos de qualificação (hex)	Códigos de função (dec)	Códigos de qualificação (hex)
41	3	Saída analógica - Single-prec flt-pt	3 (select) 4 (operar) 5 (op. direta) 6 (dir. op, no ack)	17, 27, 28 (índice)	129 (resposta)	eco da solicitação
50	1	Hora e data - hora absoluta	1 (leitura)	07 (quantidade limitada = 1)	129 (resposta)	07 (quantidade limitada) (qtd = 1)
			2 (gravação)	07 (quantidade limitada = 1)		
50	3	Hora e data - Hora absoluta na última hora registrada	2 (gravação)	07 (quantidade limitada = 1)		
51	1	Hora e data CTO - hora absoluta, sincronizada			129 (resposta) 130 (resp. não sol.)	07 (quantidade limitada) (qtd = 1)
51	2	Hora e data CTO - hora absoluta, não sincronizada			129 (resposta) 130 (resp. não sol.)	07 (quantidade limitada) (qtd = 1)
52	2	Atraso de tempo - ótimo			129 (resposta)	07 (quantidade limitada) (qtd = 1)
60	1	Objetos de classe - dados de classe 0	1 (leitura)	06 (sem faixa ou todos)		
60	2	Objetos de classe - dados de classe 1	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
			20 (habil. não sol.) 21 (desab. não sol.)	06 (sem faixa ou todos)		
60	3	Objetos de classe - dados de classe 2	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
60	4	Objetos de classe - dados de classe 3	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
			20 (habil. não sol.) 21 (desab. não sol.)	06 (sem faixa ou todos)		
80	1	Indicações internas - formato empacotado	1 (leitura)	00, 01 (partida/parada)	129 (resposta)	00, 01 (partida/parada)
			2 (gravação)	00 (partida/parada) Índice=7		
85	0	Protótipo de conjunto de dados	1 (leitura)	06 (sem faixa ou todos)		
85	1	Protótipo de conjunto de dados	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos) 17, 28 (índice)	129 (resposta)	5B (formato livre)
86	1	Descritor de conjunto de dados - conteúdo	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos) 17, 28 (índice)	129 (resposta)	5B (formato livre)
86	2	Descritor do conjunto de dados - características	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos) 17, 28 (índice)	129 (resposta)	5B (formato livre)
87	0	Conjunto de dados - valor presente	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos) 17, 28 (índice)		
87	1	Conjunto de dados - valor presente	1 (leitura)	00, 01 (partida/parada) 06 (sem faixa ou todos) 17, 28 (índice)	129 (resposta)	5B (formato livre)
88	0	Evento de conjunto de dados	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)		
88	1	Evento de conjunto de dados - instantâneo	1 (leitura)	06 (sem faixa ou todos) 07, 08 (quantidade limitada)	129 (resposta) 130 (resp. não sol.)	5B (formato livre)
90	1	Aplicação - identificador	16 (aplicação inicial) 17 (aplicação de partida) 18 (aplicação de parada)	06 (sem faixa ou todos) 5B (formato livre)		

Tabela 12 - Tabela de implementação para controladores Micro870 (Continuação)

Variação e grupo de objetos DNP			Solicitação DNP3 mestre pode emitir O controlador deve analisar		Resposta DNP3 mestre deve analisar O controlador pode emitir	
Núm. do grupo	Núm. Variável	Descrição	Códigos de função (dec)	Códigos de qualificação (hex)	Códigos de função (dec)	Códigos de qualificação (hex)
91	1	Status da operação solicitada			129 (resposta)	07 (quantidade limitada) (qtd = 1)
120	1	Autenticação - desafio	32 (Solicitação de autenticação)	5B (formato livre)	131 (Autor. resp)	5B (formato livre)
120	2	Autenticação - Resposta	32 (Solicitação de autenticação)	5B (formato livre)	131 (Autor. resp)	5B (formato livre)
120	3	Autenticação - Solicitação de modo agressivo	Qualquer solicitação	07 (quantidade limitada)	129 (resposta)	07 (quantidade limitada)
120	3	Autenticação - Solicitação de modo agressivo			130 (resp. não sol.)	07 (quantidade limitada)
120	4	Autenticação - Solicitação de status da chave da sessão	32 (Solicitação de autenticação)	07 (quantidade limitada)		
120	5	Autenticação - Status da chave da sessão			131 (Autor. resp)	5B (formato livre)
120	6	Autenticação - Alteração da chave da sessão	32 (Solicitação de autenticação)	5B (formato livre)		
120	7	Autenticação - Erro	33 (solicitação de autenticação, sem reconh.)	5B (formato livre)	131 (Autor. resp)	5B (formato livre)
120	7	Autenticação - Erro	1 (leitura)	06 (sem faixa ou todos)	129 (resposta)	5B (formato livre)
120	9	Autenticação - HMAC	Qualquer solicitação	5B (formato livre)	129 (resposta)	5B (formato livre)
120	9	Autenticação - HMAC			130 (resp. não sol.)	5B (formato livre)
Sem objeto (somente código de função)			13 (reinício frio)			
Sem objeto (somente código de função)			14 (warm restart)			
Sem objeto (somente código de função)			23 (delay meas.)			
Sem objeto (somente código de função)			24 (record current time)			

Execução de programa no Micro800

Esta seção fornece uma breve visão geral da execução de programas com um controlador Micro800.

IMPORTANTE Esta seção descreve de forma geral a execução de programas nos controladores Micro800. Certos elementos podem não ser aplicáveis ou verdadeiros para certos modelos (por exemplo, o Micro820 não suporta o controle de movimento PTO).

Para obter informações detalhadas sobre diagramas ladder, instruções, blocos de funções e assim por diante, consulte o Manual de referência: Instruções gerais para controladores programáveis Micro800, publicação [2080-RM001](#).

Visão geral da execução de programas

Uma varredura ou ciclo do Micro800 consiste em leitura de entradas, execução de programas em ordem sequencial, atualização de saídas e realização de manutenção (registro de dados, receitas, comunicações).

Os nomes dos programas devem começar com uma letra ou sublinha, seguidos de até 127 letras, dígitos ou sublinhas simples. Use linguagens de programação como lógica ladder, diagramas de bloco de funções e texto estruturado.

Até 256 programas podem ser incluídos em um projeto, dependendo da memória de controlador disponível. Por padrão, os programas são cíclicos (executados uma vez por ciclo ou varredura). À medida que cada novo programa é adicionado ao projeto, atribui-se a ele o próximo número de ordem consecutivo. Quando você inicia o Project Organizer no Connected Components Workbench, ele exibe os ícones de programa com base nessa ordem. Você pode visualizar e modificar o número de ordem de um programa a partir das propriedades do programa. Contudo, o Project Organizer não mostra a nova ordem até a próxima vez em que o projeto é aberto.

O controlador Micro800 é compatível com saltos dentro de um programa. Chame uma sub-rotina de código dentro de um programa encapsulando o código como uma Função definida pelo usuário (UDF) ou bloco de funções definido pelo usuário (UDFB). Um UDF é similar a uma subrotina tradicional e usa menos memória que um UDFB, enquanto um UDFB pode ter múltiplas instâncias. Ainda que um UDFB possa ser executado dentro de outro UDFB, uma profundidade máxima de encadeamento de cinco é suportada. Se ela for excedida, ocorre um erro de compilação. Isso também se aplica a UDFs.

De modo alternativo, você pode atribuir um programa a uma interrupção disponível e fazê-lo ser executado somente quando a interrupção é disparada.

Um programa atribuído à rotina de falha do usuário roda uma vez logo antes de o controlador entrar em modo de falha.

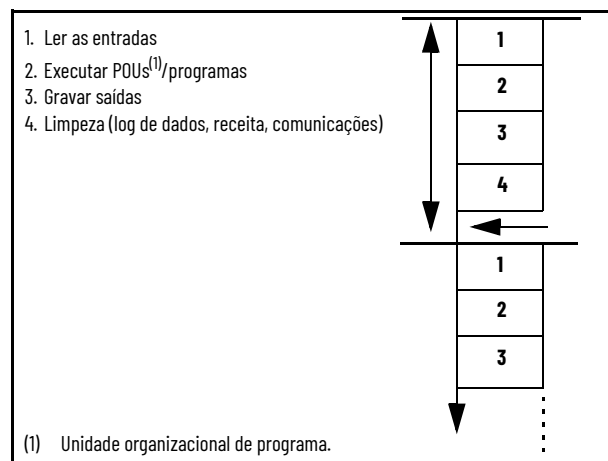
Além da rotina de falha do usuário, os controladores Micro800 também suportam duas interrupções selecionadas em função do tempo (STI). As STIs executam programas atribuídos uma vez a cada intervalo de valor de referência (1 a 65.535 ms).

As variáveis globais do sistema associadas com ciclos/varreduras são:

- __SYSVA_CYCLECNT – Contador de ciclo
- __SYSVA_TCYCURRENT – Tempo de ciclo atual
- __SYSVA_TCYMAXIMUM – Tempo de ciclo máximo desde a última partida.

Regras para execução

Esta seção ilustra a execução de um programa. A execução segue quatro passos principais dentro de uma malha. A duração da malha é um tempo de ciclo para um programa.



Quando um tempo de ciclo é especificado, um recurso espera até que esse tempo tenha transcorrido antes de iniciar a execução de um novo ciclo. O tempo de execução das POUs varia dependendo do número de instruções ativas. Quando um ciclo excede o tempo especificado, a malha continua a executar o ciclo, mas define um flag de superposição. Nesse caso, a aplicação não roda mais em tempo real.

Quando um tempo de ciclo não é especificado, um recurso realiza todas as etapas na malha e então reinicia um novo ciclo sem esperar.

Módulo opcional

Normalmente, antes da etapa de leitura de entradas, o controlador verificará a presença de quaisquer módulos de plug-in e de expansão de E/S configurados. Se um plug-in ou módulo de E/S de expansão estiver faltando, o controlador entrará em falha. No software Connected Components Workbench release 10 ou posterior, uma opção de configuração de módulo opcional é adicionada para evitar que um plug-in E/S ou módulo de E/S de expansão faltante faça o controlador entrar em falha se habilitado. Esta opção pode ser habilitada separadamente para cada plug-in E/S ou módulo de E/S de expansão.



ATENÇÃO: Se estiver habilitado o recurso de módulo opcional, use a instrução MODULE_INFO para verificar que o módulo está presente porque o controlador não apresentará falha se o módulo estiver ausente.

Considerações sobre carga e desempenho do controlador

Dentro de um ciclo de varredura de programa, a execução das etapas principais (como indicado no Diagrama de regras de execução) pode ser interrompida por outras atividades do controlador, que tenham uma prioridade mais alta que as etapas principais. Essas atividades incluem,

1. Eventos interrompidos pelo usuário, incluindo interrupções STI, EII e HSC (quando aplicável);
2. Recepção e transmissão de pacote de dados de comunicação;
3. Execução periódica do motor de controle de movimento PTO (se compatível com o controlador).

Quando uma ou diversas dessas atividades ocupam uma porcentagem significativa do tempo de execução do controlador Micro800, o tempo de ciclo de varredura do programa será prolongado. A falha de limite de tempo watchdog (0xDo11) pode ser relatada se o impacto dessas atividades for subestimado, e o limite de tempo watchdog é determinado marginalmente. A configuração do Watchdog ajusta por padrão como 2 s e geralmente não precisa ser alterada.

Execução periódica de programas

Para aplicações em que a execução periódica de programas com temporização precisa é necessária, tais como PID, é recomendado que a STI (interrupção selecionada em função do tempo) seja utilizada para executar o programa. A STI fornece intervalos de tempo precisos.

Não é recomendado que a variável do sistema __SYSVA_TCYCYCTIME seja usada para executar todos os programas periodicamente, pois isso faz com que todas as comunicações sejam executadas a essa taxa.



ADVERTÊNCIA: Os limites de tempo de comunicação podem ocorrer se o tempo de ciclo programado for definido como muito lento (por exemplo, 200 ms) para manter as comunicações.

Variável de sistema para tempo de ciclo programado

Variável	Type [Tipo]	Descrição
__SYSVA_TCYCYCTIME	TIME	Tempo de ciclo programado. Nota: O tempo de ciclo programado somente aceita valores em múltiplos de 10 ms. Se o valor inserido não for um múltiplo de 10, ele será arredondado para cima para o próximo múltiplo de 10.

Energização e primeira varredura

No modo de programa, todas as variáveis de entrada digital e analógica mantêm seu último estado, e os LEDs são sempre atualizados. Todas as variáveis de saída digitais e analógicas também mantêm seu último estado, mas somente as saídas analógicas mantêm seu último estado enquanto as saídas digitais estão desligadas.

Quando se faz a transição de modo de programa para modo de operação, todas as variáveis de saída analógica mantêm seu último estado, mas todas as variáveis de saída digital são descarregadas.

Duas variáveis de sistema também estão disponíveis a partir da revisão 2 e posteriores:

Variáveis do sistema para varredura e energização na versão do firmware 2 e posteriores

Variável	Type [Tipo]	Descrição
_SYSVA_FIRST_SCAN	BOOL	Primeiro bit de varredura. Pode ser utilizado para inicializar ou restaurar variáveis imediatamente após cada transição de modo Programa para Operação. Observação: Verdadeiro apenas para a primeira varredura. Após isso, é falso.
_SYSVA_POWER_UP_BIT	BOOL	Bit de energização. Pode ser utilizado para inicializar ou restaurar variáveis imediatamente após o download, a partir do software Connected Components Workbench ou imediatamente após ser carregado, a partir do módulo de backup da memória (por exemplo, cartão microSD™). Observação: Verdadeiro apenas para a primeira varredura após uma energização ou execução de uma nova lógica ladder pela primeira vez.

Retenção de variável

Após ligar e desligar, todas as variáveis dentro de instâncias de instruções são apagadas. Os controladores Micro830 e Micro850 retêm todas as variáveis criadas pelo usuário. Os controladores Micro810® e Micro820 somente podem reter um máximo de 400 bytes de valores de variável criados pelo usuário. Os controladores Micro870 e somente podem reter um máximo de 128 kilobytes de valores de variável criados pelo usuário.

Por exemplo: uma variável criada pelo usuário chamada My_Timer do tipo de dados Time será retida após ligar e desligar a alimentação, porém o tempo decorrido (ET) criada pelo usuário na instrução de temporizador TON será apagada. Isso significa que, após um ciclo de energia, as variáveis globais são limpas ou configuradas para o valor inicial e, dependendo do controlador, alguns ou todos os valores de variáveis criados pelo usuário são retidos. Você pode escolher quais variáveis reter selecionando-as na página de variáveis globais.

Name	Data Type	Dimension	String Size	Initial Value	Attribute	Retained
Retained_Boolean	BOOL				Read/Write	<input checked="" type="checkbox"/>

Alocação de memória

Dependendo do tamanho da base, a memória disponível nos controladores Micro800 é mostrada na tabela abaixo.

Alocação de memória para controladores Micro800

Atributo	10/16 pontos (Micro830)	20 pontos (Micro820)	24 e 48 pontos (Micro830, Micro850)	24 pontos (Micro870)
Etapas do programa ⁽¹⁾	4 K	10 K	10 K	20 K
Bytes de dados	8 KB	20 KB	20 KB	40 KB

(1) O tamanho dos dados e o programa estimado são "típicos" – as etapas do programa e as variáveis são criadas dinamicamente.
1 etapa do programa = 12 bytes de dados.

Estas especificações para instrução e tamanho de dados são números típicos. Quando um projeto é criado para Micro800, a memória é alocada dinamicamente como programa ou memória de dados no tempo de construção. Isso significa que o tamanho do programa pode exceder as especificações publicadas se o tamanho dos dados for sacrificado e vice-versa. Essa flexibilidade permite um uso máximo da memória de execução. Além das variáveis definidas pelo usuário, a memória de dados também contém quaisquer constantes e variáveis temporárias geradas pelo compilador em tempo de construção.

Se o projeto do usuário for maior, o tempo de ativação será afetado. O tempo de inicialização típico é de 10 a 15 segundos para todos os controladores. No entanto, se o seu projeto tiver muitos valores iniciais e de projeto, isso poderá fazer com que o tempo de ativação exceda 30 segundos. Após o boot, as conexões Ethernet/IP podem levar até 60 segundos para serem estabelecidas.

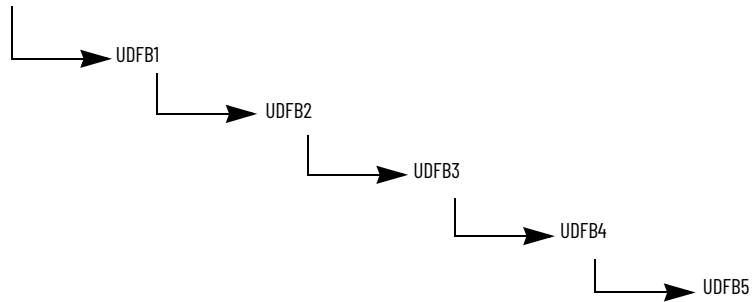
Os controladores Micro800 têm memória de projeto, que armazena uma cópia do projeto inteiro do que se fez download (inclusive comentários), assim como uma memória de configuração para armazenar informações de ajuste plug-in, etc.

Orientações e limitações para usuários avançados

Seguem algumas orientações e limitações que devem ser consideradas quando se programa um controlador Micro800 usando o software Connected Components Workbench:

- Cada programa/POU pode usar até 64 Kb de espaço de endereço interno. Para todos os controladores, exceto o Micro870, é recomendado que você divida programas grandes em programas menores para melhorar a legibilidade do código, simplificar a depuração e as tarefas de manutenção.
- Uma Função definida pelo usuário (UDF) usa significativamente menos memória que um Bloco de funções definido pelo usuário (UDFB). Por exemplo, 30% menos para um programa de tamanho típico em comparação com um UDFB para uma instância. A economia aumenta à medida que aumenta o número de instâncias UDFB.
- Um Bloco de funções definido pelo usuário (UDFB) pode ser executado dentro de outro UDFB, com um limite de cinco UDFBs encadeados. Evite criar UDFBs com referências a outros UDFBs, pois executar esses UDFBs muitas vezes pode resultar em um erro de compilação. Isso também se aplica a UDFs.

Exemplo de cinco UDFBs encadeados



- O texto estruturado (ST) é muito mais eficiente e fácil de usar que a lógica ladder, quando usado para equações. Se você está acostumado a usar a instrução RSLogix 500® CPT Compute, uma ótima alternativa é usar ST combinado com UDF ou UDFB. Como exemplo, para um cálculo de relógio astronômico, o texto estruturado usa 40% menos instruções.

Display_Output LD:

Uso de memória (código): 3148 etapas

Uso de memória (dados): 3456 bytes

Display_Output ST:

Uso de memória (código): 3148 etapas

Uso de memória (dados): 3456 bytes

- Você pode encontrar um erro de memória reservada insuficiente enquanto realizar um download e compilar um programa acima de um determinado tamanho. Uma solução temporária é usar vetores, especialmente se houver muitas variáveis.

Controle de movimento

Geralmente, dois tipos de controle de movimento são usados em aplicações de movimento do controlador Micro800, que têm servo-drives Kinetix® 3.

- Movimento indexado - Micro800 distribui índices de posição para o servo-drive usando comunicações Modbus RTU ou E/S digital. Usado para posicionamento simples. Consulte a publicação [CC-QSo25](#) para exemplo de elemento fundamental.
- Movimento PTO – O controlador Micro800 usa saídas de pulso e direção para o servo-drive para um controle preciso da posição e velocidade com comunicações Modbus RTU ou E/S digital para realimentação. A configuração e as instruções de controle de movimento do Micro800 facilitam a programação. Consulte a publicação [CC-QSo33](#) para exemplo de elemento fundamental.

Controle de movimento PTO

Certos controladores Micro830, Micro850 e Micro870, mostradas na [Tabela 13](#), suportam controle de movimento via saídas de trem de pulso (PTO) de alta velocidade. A funcionalidade PTO se refere à capacidade de um controlador de gerar com precisão um número específico de pulsos a uma frequência especificada. Esses pulsos são enviados para um dispositivo de controle de movimento, como um servo-drive, que por sua vez controla o número de rotações (posição) de um servomotor. Cada PTO é mapeado exatamente a um eixo, para permitir controle de movimento simples em motores de passo e servodrives com entrada de direção/pulso.

Como o ciclo de trabalho da PTO pode ser substituído dinamicamente, a PTO também pode ser usada como saída da modulação por largura de pulso (PWM).

Os eixos de movimento e PTO/PWM suportados nos controladores Micro830 e Micro850 são listados abaixo.

Tabela 13 - Eixo de movimento e PTO/PWM são compatíveis com o Micro830, Micro850 e Micro870

Controlador	PTO (incorporado)	Número de eixos suportados
10/16 pontos ⁽¹⁾ 2080-LC30-10QVB 2080-LC30-16QVB	1	1
24 pontos 2080-LC50-24QVB ⁽¹⁾ 2080-LC50-24QBB ⁽¹⁾ 2080-LC50-24QVB 2080-L50E-24QVB 2080-LC50-24QBB 2080-L50E-24QBB 2080-LC70-24QBB 2080-L70E-24QBB 2080-LC70-24QBBK 2080-L70E-24QBBK 2080-L70E-24QBBN	2	2
48 pontos 2080-LC30-48QVB ⁽¹⁾ 2080-LC30-48QBB ⁽¹⁾ 2080-LC50-48QVB 2080-L50E-48QVB 2080-LC50-48QBB 2080-L50E-48QBB	3	3

(1) Para catálogos Micro830, funcionalidade de saída de trem de pulso é suportada somente a partir da revisão 2 do firmware ou mais recente.



ATENÇÃO: Para utilizar o recurso de movimento do Micro800 com eficiência, usuários precisam possuir um conhecimento básico do seguinte:

- Componentes e parâmetros da PTO
Consulte [Use o recurso de controle de movimento do Micro800 na página 145](#) para uma visão geral dos componentes de movimento e suas relações.
- Programar e trabalhar com elementos no software Connected Components Workbench
O usuário precisa estar familiarizado com a programação de diagrama de lógica ladder, texto estruturado ou diagrama de blocos de funções para trabalhar com blocos de função de posicionamento, variáveis e parâmetros de configuração do eixo.



ATENÇÃO: Para aprender mais sobre o software Connected Components Workbench e descrições detalhadas das variáveis para os blocos de função de posicionamento, você pode consultar a ajuda on-line do software Connected Components Workbench que acompanha a instalação do software Connected Components Workbench.

IMPORTANTE A função PTO somente pode ser utilizada com as E/S incorporadas do controlador. Ela não pode ser usada com módulos de E/S de expansão.

Use o recurso de controle de movimento do Micro800

O recurso de controle de movimento do Micro800 possui os elementos a seguir. Novos usuários precisam ter um entendimento básico da função de cada elemento para utilizar com eficiência o recurso.

Componentes do Controle de movimento

Elemento	Descrição	Página
Saídas de trem de pulso	Consiste de uma saída de pulso e uma saída de direção. Uma interface padrão para controlar um servo-inversor ou inversor passo a passo.	<ul style="list-style-type: none"> • Sinais de entrada e saída na página 146
Eixo	Por um ponto de vista de sistema, um eixo é um equipamento mecânico movido por uma combinação de motor e inversor. O inversor recebe comandos de posição por meio da interface de saídas de trem de pulso do Micro800, segundo a execução do CLP dos blocos de função de posicionamento. No controlador Micro800, é uma saída de trem de pulso e um conjunto de entradas, saídas e configuração.	<ul style="list-style-type: none"> • Movimento do eixo e parâmetros na página 158 • Configuração do eixo de movimento no Connected Components Workbench na página 169
Blocos de funções de movimento	Um conjunto de instruções que configuram ou agem sobre um eixo de movimento.	<ul style="list-style-type: none"> • Ajuda online do Connected Components Workbench • Blocos de funções de controle de movimento na página 149 • Tipo de dados Axis.Ref na página 164 • Códigos de erro de bloco de funções e status de eixo na página 166 • Bloco de funções de retorno à posição inicial na página 181
Variação da aceleração	Taxa de variação da aceleração. O componente de variação da aceleração é relevante principalmente no início e no final do movimento. Uma variação de aceleração muito alta pode induzir vibrações.	<ul style="list-style-type: none"> • Consulte Entradas Acceleration, Deceleration e Jerk na página 151.

Para utilizar o recurso de controle de movimento do Micro800, você precisa:

1. Configurar as propriedades do eixo.
Consulte [Configuração do eixo de movimento no Connected Components Workbench na página 169](#) para obter instruções.
2. Escrever seu programa de controle de movimento por meio do software Connected Components Workbench.
Para instruções sobre como usar o recurso de controle de movimento do Micro800, consulte as instruções de início rápido, Use the Motion Control Feature on Micro800 Controllers, publicação [2080-QS001](#).
3. Conexão da fiação do controlador
Para entradas/saídas fixas e configuráveis, consulte [Sinais de entrada e saída na página 146](#). Consulte [Exemplo de configuração de fiação de movimento em 2080-LC30-xxQVB/2080-LC50-xxQVB/2080-LC70-xxQVB na página 148](#) para mais referências.

As próximas seções fornecem uma descrição mais detalhada dos componentes de movimento. Você também pode consultar a ajuda on-line do software Connected Components Workbench para mais informações sobre cada bloco de funções de movimento e suas entradas e saídas variáveis.

Sinais de entrada e saída

Sinais de controle de entrada/saída múltiplos são exigidos para cada eixo de controle de movimento, como descrito nas próximas tabelas. Pulso PTO e direção PTO são exigidos para um eixo. O resto das entrada/saídas podem ser desabilitadas e reutilizadas como E/S comum.

Entrada/saída de PTO fixa

Sinais de movimento	PT00 (EM_00)		PT01 (EM_01)		PT02 (EM_02)	
	Nome lógico no software	Nome no borne	Nome lógico no software	Nome no borne	Nome lógico no software	Nome no borne
Pulso PTO	_IO_EM_DO_00	O-00	_IO_EM_DO_01	O-01	IO_EM_DO_02	O-02
Direção PTO	_IO_EM_DO_03	O-03	_IO_EM_DO_04	O-04	IO_EM_DO_05	O-05
Chave de fim de curso inferior (negativo)	_IO_EM_DI_00	I-00	_IO_EM_DI_04	I-04	IO_EM_DI_08	I-08
Chave de fim de curso superior (positivo)	_IO_EM_DI_01	I-01	_IO_EM_DI_05	I-05	IO_EM_DI_09	I-09
Chave de posição inicial absoluta	_IO_EM_DI_02	I-02	_IO_EM_DI_06	I-06	IO_EM_DI_10	I-10
Chave de entrada do sensor de toque	_IO_EM_DI_03	I-03	_IO_EM_DI_07	I-07	IO_EM_DI_11	I-11

Entradas/saídas configuráveis

Sinais de movimento	Entrada/Saída	Observações
Servo/inversor energizado	SAÍDA	Pode ser configurado como qualquer saída incorporada.
Servo/inversor pronto	ENTRADA	Pode ser configurado como qualquer entrada incorporada.
Sinal de "em posição" (do servo/motor)	ENTRADA	Pode ser configurado como qualquer entrada incorporada.
Marcador de posição inicial	ENTRADA	Pode ser configurada como qualquer entrada incorporada, da entrada 0 à 15.

Essas E/S podem ser configuradas por meio do recurso de configuração do eixo no Connected Components Workbench. Quaisquer saídas atribuídas para movimento não devem ser controladas no programa do usuário.

Consulte [Configuração do eixo de movimento no Connected Components Workbench na página 169](#).

IMPORTANTE

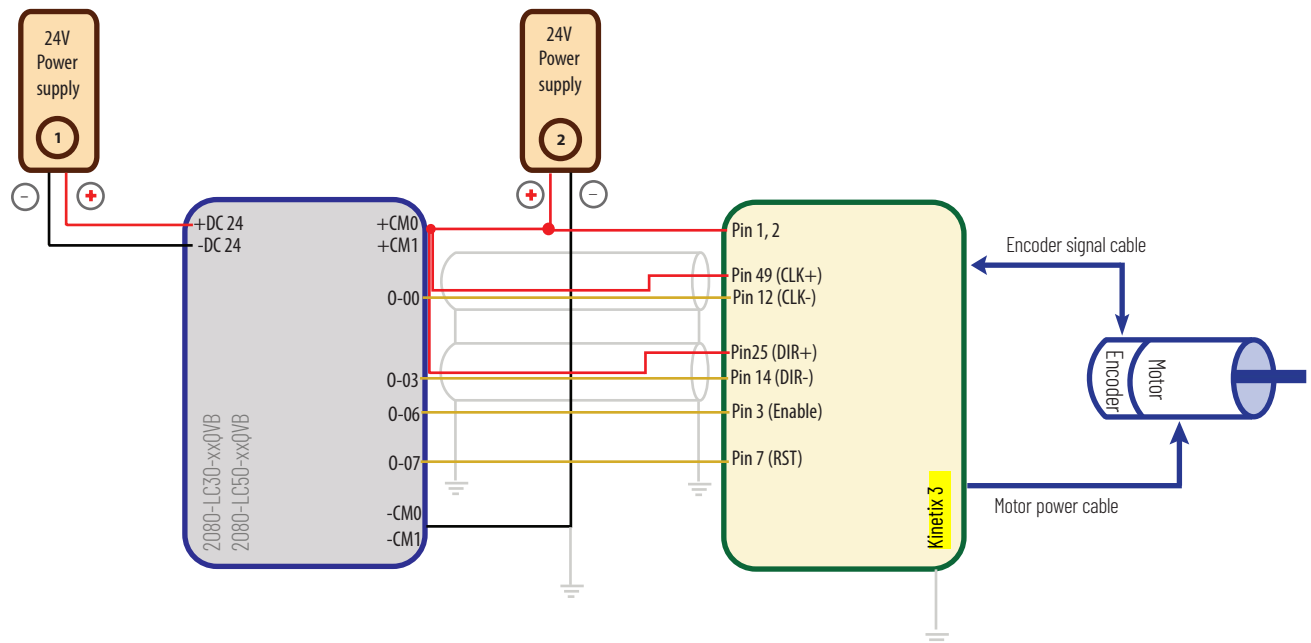
Se uma saída é configurada para movimento, então essa saída não pode mais ser controlada ou monitorada pelo programa do usuário e não pode ser forçada. Por exemplo, quando uma saída de pulso PTO está gerando pulsos, a variável lógica correspondente IO_EM_DO_xx não alternará o seu valor e não exibirá os pulsos no monitor de variáveis, mas o LED físico dará uma indicação.

Se uma entrada estiver configurada para movimento, então forçar a entrada afeta somente a lógica do programa de usuário, e não a de movimento. Por exemplo, se a entrada inversor pronto for falsa, então o usuário não pode forçar inversor pronto a ser verdadeira forçando a variável lógica IO_EM_DI_xx correspondente a ser verdadeira.

Descrição de entrada/saída de fiação de movimento

Sinais de movimento	Entrada/Saída	Descrição	Singularidade
Pulso PTO	SAÍDA	Pulso PTO a partir da saída rápida incorporada, para ser conectado à entrada PTO do inversor.	Não compartilhada
Direção PTO	SAÍDA	Indicação de direção de pulso PTO, a ser conectada à entrada de direção do inversor.	Não compartilhada
Servo/inversor energizado	SAÍDA	O sinal de controle utilizado para ativar/desativar o servo/inversor. Esse sinal fica ativo quando se comanda MC.Power (energizado).	Pode ser compartilhado com mais de um inversor
Chave de fim de curso inferior (negativo)	ENTRADA	A entrada para chave de fim de curso negativa de hardware, a ser conectada a sensor de limite negativo elétrico/mecânico.	Não compartilhada
Chave de fim de curso superior (positivo)	ENTRADA	A entrada para chave de fim de curso positiva de hardware, a ser conectada a sensor de limite positivo elétrico/mecânico.	Não compartilhada
Chave de posição inicial absoluta	ENTRADA	A entrada para chave de posição inicial de hardware (sensor), a ser conectada a sensor de posição inicial mecânico/elétrico.	Não compartilhada
Chave de entrada do sensor de toque	ENTRADA	A entrada para sinal de sensor de toque de hardware, a ser usada com os blocos de função Motion MC.TouchProbe e MC.AbortTrigger para capturar a posição comandada do eixo durante o percurso do movimento.	Não compartilhada
Servo/inversor pronto	ENTRADA	O sinal de entrada que indica que o servo/inversor está pronto para receber pulso PTO e sinal de direção do controlador. Nenhum bloco de funções móvel pode ser atribuído a um eixo antes que este tenha esse sinal de entrada pronto, se esse sinal estiver habilitado na página de propriedades do eixo ou configuração de eixo de movimento.	Pode ser compartilhado com mais de um inversor
Sinal de "em posição" (do servo/motor)	ENTRADA	O sinal de entrada que indica que a peça móvel está na posição comandada. Esse sinal precisa estar ativo após a peça móvel atingir a posição comandada para os blocos de funções MoveAbsolute e MoveRelative. Para os blocos de funções MoveAbsolute e MoveRelative, quando In.Position estiver habilitado, o controlador relatará um erro (EP_MC_MECHAN_ERR) se o sinal não estiver ativo em até 5 segundos quando o último pulso PTO for enviado.	Não compartilhada
Marcador de posição inicial	ENTRADA	Esse sinal é o sinal de pulso zero do encoder do motor. Esse sinal pode ser utilizado para ajuste-fino da sequência de movimento para posição inicial para aprimorar a precisão do deslocamento para essa posição.	Não compartilhada

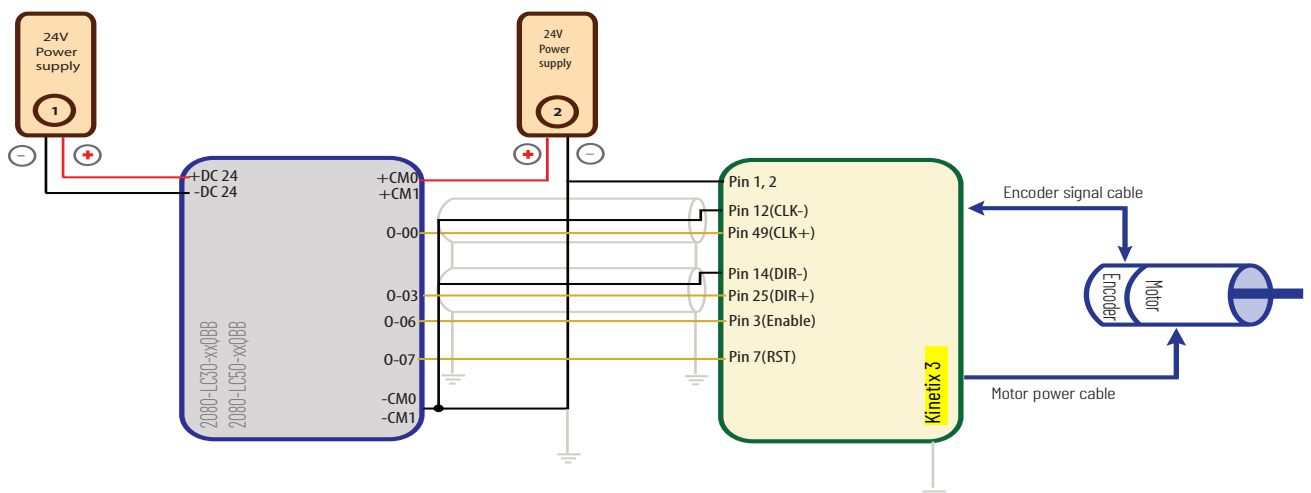
Exemplo de configuração de fiação de movimento em 2080-LC30-xxQVB/2080-LC50-xxQVB/2080-LC70-xxQVB



Observações:

1. A habilitação do inversor (pino 3) e o reset do inversor (pino 7) estarão operando como entradas de fornecimento quando (pinos 1, 2) conectadas a da fonte de alimentação 2.
2. Para ajudá-lo a configurar os parâmetros do inversor Kinetix3 para que o inversor possa se comunicar e ser controlado por um controlador Micro830/Micro850/Micro870, consulte a publicação [CC-QS033](#). O parâmetro Tipo de Comando deve ser definido para "Step/Direction.Positive Logic", e o parâmetro Tipo de Saída do Controlador deve ser definido para "Open Collector Input".

Exemplo de configuração de fiação de movimento em 2080-LC30-xxQBB/2080-LC50-xxQBB/2080-LC70-xxQBB



Observações:

1. A habilitação de inversor (pino 3) e o reset do inversor (pino 7) estarão operando como entradas de consumo quando (pinos 1, 2) conectadas a da fonte de alimentação 2.
2. Para ajudá-lo a configurar os parâmetros do inversor Kinetix3 para que o inversor possa se comunicar e ser controlado por um controlador Micro830/Micro850/Micro870, consulte a publicação [CC-QS033](#). O parâmetro Tipo de Comando deve ser definido para "Step/Direction.Positive Logic", e o parâmetro Tipo de Saída do Controlador deve ser definido para "Open Collector Input".

Blocos de funções de controle de movimento

Blocos de funções de controle de movimento instruem um eixo a uma posição, distância, velocidade e estado especificados.

Blocos de funções são classificados como de movimento (geração de movimento) e administrativos.

Blocos de funções administrativas

Nome do bloco de funções	Nome do bloco de funções
MC_Power	MC_ReadAxisError
MC_Reset	MC_ReadParameter
MC_TouchProbe	MC_ReadBoolParameter
MC_AbortTrigger	MC_WriteParameter
MC_ReadStatus	MC_WriteBoolParameter
MC_SetPosition	



ADVERTÊNCIA: Durante a alteração do modo de operação (RMC), o bloco de funções MC_Power deve ser desabilitado, o que removerá a alimentação do eixo. Do contrário, o eixo permanecerá energizado mesmo se o bloco de funções for apagado.

Tome nota do seguinte:

- Se uma nova instância de MC_Power acessa o eixo, este entrará no estado de parada por erro.
- Se MC_Power estiver dentro de UDFB e qualquer edição for feita no UDFB que altere o gabarito do UDFB (por exemplo, adição de uma variável local), o eixo entrará no estado de parada por erro.

Blocos de funções de movimento

Nome do bloco de funções	Descrição	Estado correto do eixo para atribuição de bloco de funções
MC_MoveAbsolute	Esse bloco de funções comanda um eixo a uma posição absoluta especificada.	Standstill, Discrete Motion, Continuous Motion
MC_MoveRelative	Este bloco de funções comanda um eixo de uma distância especificada relativa à posição real no momento da execução.	Standstill, Discrete Motion, Continuous Motion
MC_MoveVelocity	Este bloco de funções comanda um eixo sem fim a se mover a uma velocidade especificada.	Standstill, Discrete Motion, Continuous Motion
MC_Home	Este bloco de funções comanda o eixo a realizar a sequência de "buscar posição inicial". A entrada "Position" é utilizada para configurar a posição absoluta quando o sinal de referência é detectado, e o offset inicial configurado é atingido. Este bloco de funções é concluído em "StandStill" se a sequência de movimento para posição inicial for bem-sucedida.	Standstill

Blocos de funções de movimento (Continuação)

Nome do bloco de funções	Descrição	Estado correto do eixo para atribuição de bloco de funções
MC_Stop	Este bloco de funções comanda uma parada de eixo e transfere o eixo para o estado "Stopping". Ele aborta qualquer execução de bloco de funções em andamento. Enquanto o eixo estiver no estado Stopping, nenhum outro bloco de funções poderá realizar qualquer movimento no mesmo eixo. Após o eixo atingir velocidade zero, a saída Done é configurada para TRUE imediatamente. O eixo permanece no estado "Stopping" enquanto Execute ainda for TRUE ou a velocidade zero ainda não tiver sido atingida. Assim que "Done" for DEFINIDO e "Execute" for FALSE, o eixo irá para o estado "StandStill".	Standstill, Discrete Motion, Continuous Motion, Homing
MC_Halt	Este bloco de funções comanda um eixo a uma parada de movimento controlada. O eixo é movido para o estado "Discrete Motion", até que a velocidade seja zero. Com a saída Done definida, o estado é transferido para "StandStill".	Standstill, Discrete Motion, Continuous Motion



ATENÇÃO: Durante a alteração do modo de operação, os blocos de função de movimento somente podem ser apagados quando o bloco de funções foi realizado ou abortado. Do contrário, pode ocorrer um comportamento não pretendido do eixo e do bloco de funções.



ATENÇÃO: Cada bloco de funções de movimento possui um conjunto de entradas e saídas variáveis que permite que você controle uma instrução de movimento específica. Consulte a ajuda on-line do Connected Components Workbench para uma descrição dessas entradas e saídas variáveis.

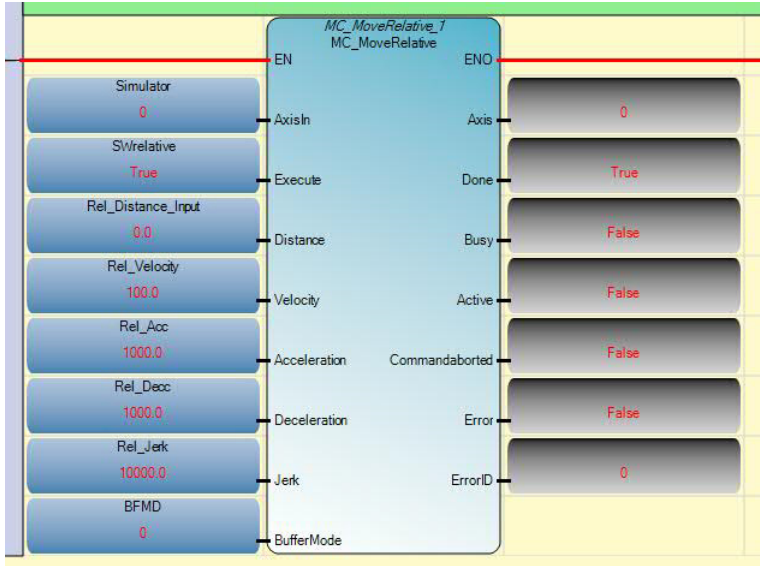
Regras gerais para os Blocos de funções de controle de movimento

Para trabalhar com os blocos de funções de controle de movimento, os usuários precisam estar familiarizados com as regras gerais a seguir.

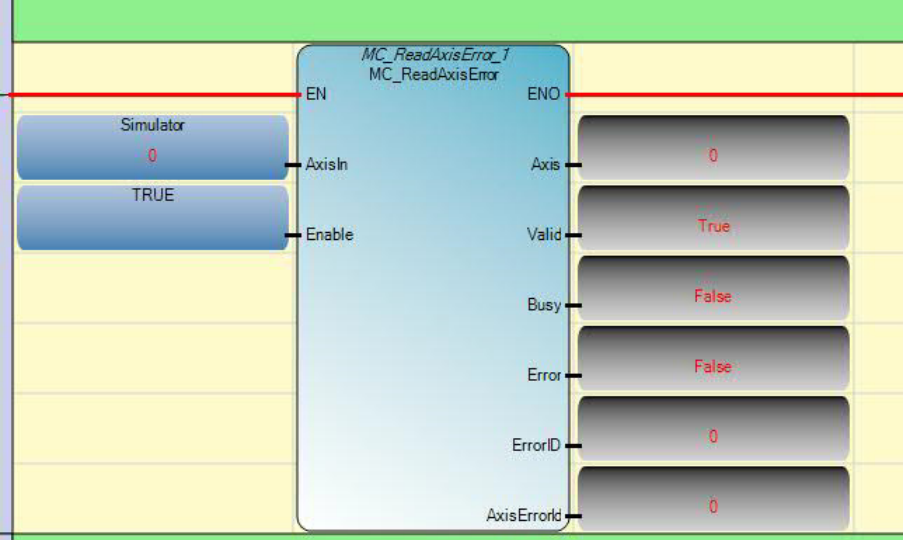
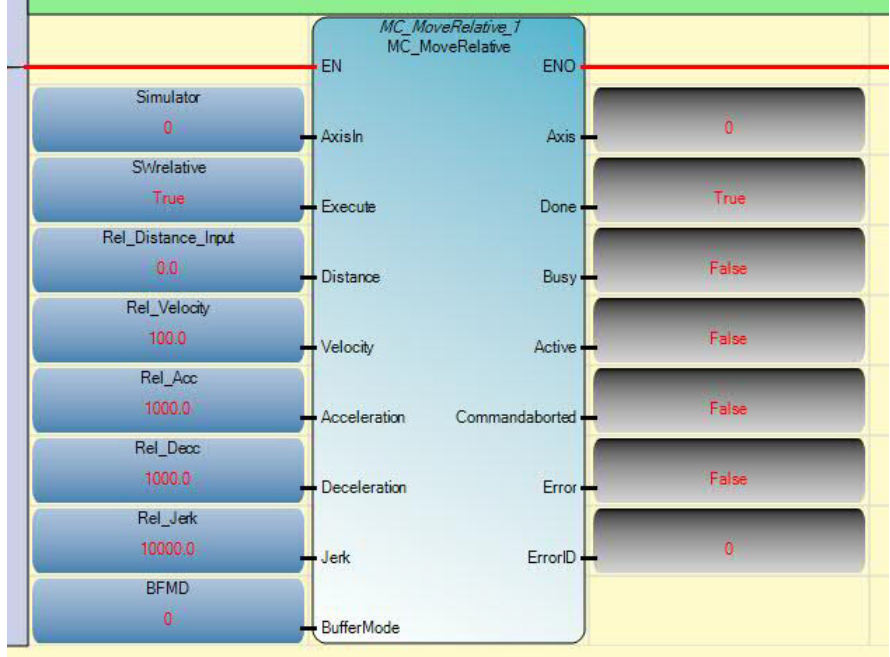
Regras gerais para o bloco de funções de movimento

Parâmetro	Regras gerais
Parâmetros de entrada	Quando Executar é verdadeiro: Os parâmetros são usados com a borda de subida da entrada de Execução. Para modificar qualquer parâmetro, é necessário mudar o(s) parâmetro(s) de entrada e gerar movimento novamente. Quando Habilitar é verdadeiro: Os parâmetros são utilizados com a borda de subida da entrada Enable e podem ser modificados continuamente.
Entradas excedendo os limites da aplicação	Se um bloco de funções é configurado com parâmetros que resultam em uma violação de limites de aplicação, a instância do bloco de funções gera um erro. Nesse caso, a saída Error será marcada como energizada e a informação do erro será indicada pela saída ErrorID. O controlador, na maioria dos casos, permanecerá no modo de operação, e nenhum erro de movimento será reportado como uma falha grave de controlador.
Entrada de posição/distância	Para o bloco de funções MC_MoveAbsolute, a entrada de posição é a localização absoluta comandada para o eixo. Para MC_MoveRelative, a entrada de distância é a localização relativa (considerando a posição atual do eixo como 0) da posição atual.
Entrada de velocidade	Velocity pode ser um valor assinado. Recomenda-se aos usuários utilizar velocidade positiva. Entrada de direção para o bloco de funções MC_MoveVelocity pode ser utilizada para definir a direção do movimento (ou seja, velocidade negativa x direção negativa = velocidade positiva). Para os blocos de funções MC_MoveRelative e MC_MoveAbsolute, o valor absoluto da velocidade é utilizado. A entrada Velocity não precisa ser atingida se a entrada Jerk for igual a 0.

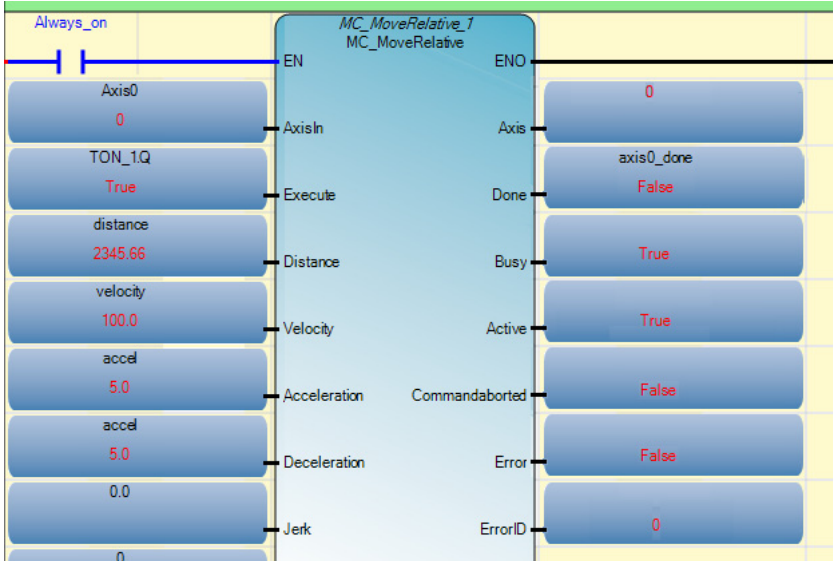
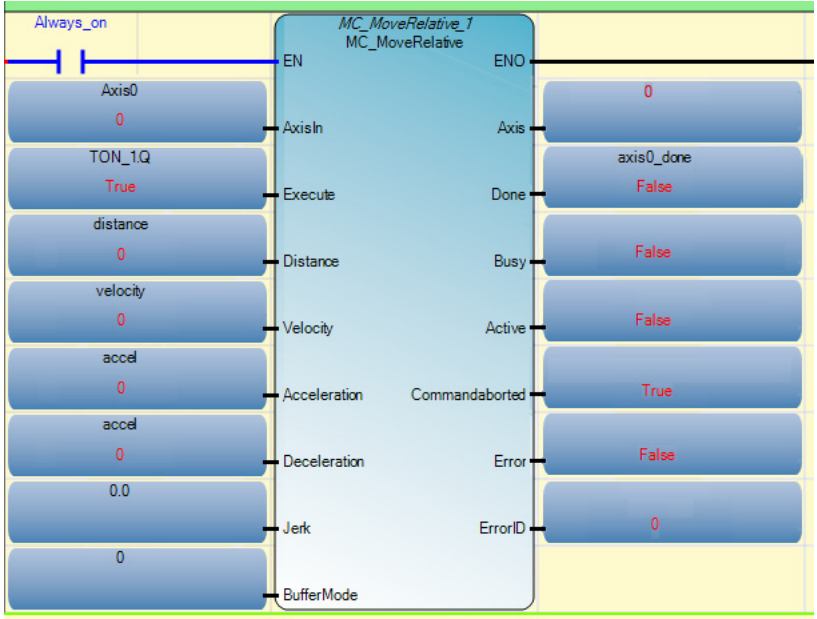
Regras gerais para o bloco de funções de movimento (Continuação)

Parâmetro	Regras gerais
Entrada de direção	<p>Para MC_MoveAbsolute, a entrada de direção é ignorada. (isso é reservado para uso futuro)</p> <p>Para MC_MoveVelocity, o valor da entrada de direção pode ser 1 (direção positiva), 0 (direção atual) ou -1 (direção negativa). Para qualquer outro valor, somente o sinal é levado em consideração. Por exemplo, -3 determina direção negativa, +2 determina direção positiva, e assim sucessivamente.</p> <p>Para MC_MoveVelocity, o sinal resultante do valor do produto derivado de velocidade x direção decide a direção do posicionamento, se o valor não for 0. Por exemplo, se velocidade x direção = +300, então a direção é positiva.</p>
Entradas Acceleration, Deceleration e Jerk	<ul style="list-style-type: none"> As entradas Deceleration ou Acceleration devem possuir um valor positivo. Se Deceleration ou Acceleration estiver configurada para ser um valor não positivo, um erro será reportado (ID do erro: MC_FB_ERR_RANGE). A entrada Jerk deve possuir um valor não negativo. Se Jerk estiver configurada para ser um valor negativo, um erro será reportado. (ID do erro: MC_FB_ERR_RANGE). Se Jerk máxima estiver configurada como zero na configuração de movimento do Connected Components Workbench, todos os parâmetros para o bloco de funções de movimento precisam ser configurados como zero. Caso contrário, o bloco de funções reporta um erro (ID do erro: MC_FB_ERR_RANGE). Se Jerk estiver configurado como um valor diferente de zero, um perfil de curva S é gerado. Se Jerk for configurado como zero, um perfil trapezoidal é gerado. Se o motor de movimento falha ao gerar o perfil de movimento prescrito nos parâmetros de entrada dinâmicos, o bloco de funções reporta um erro (ID do erro: MC_FB_ERR_PROFILE). <p>Consulte Códigos de erro de bloco de funções e status de eixo na página 166 para obter mais informações sobre códigos de erro.</p>
Exclusividade de saída	<p>Com Executar: As saídas Busy, Done, Error e CommandAborted indicam o estado do bloco de funções e são mutuamente exclusivas – somente uma delas pode ser verdadeira em um bloco de funções. Se executar for verdadeiro, uma dessas saídas precisa ser verdadeira.</p> <p>As saídas Done, Busy, Error, ErrorID, e CommandAborted são restauradas com a bora de descida de Executar. Porém, a bora de descida de Executar não para nem mesmo influencia a execução do bloco de funções real. Mesmo se executar for resetado antes do bloco de funções ser completado, as saídas correspondentes são configuradas para ao menos um ciclo.</p> <p>Se uma instância de um bloco de funções recebe um novo comando Executar antes de ser completada (como uma série de comandos na mesma instância), o novo comando Executar é ignorado, e a instrução dada anteriormente continua com a execução.</p> 

Regras gerais para o bloco de funções de movimento (Continuação)

Parâmetro	Regras gerais
Exclusividade de saída	<p>Com Habilitar: As saídas Valid e Error indicam se um bloco de funções lido é executado com sucesso. Elas são mutuamente exclusivas: somente uma delas pode ser verdadeira em um bloco de funções para MC_ReadBool, MC_ReadParameter, MC_ReadStatus. As saídas Valid, Enabled, Busy, Error, e ErrorID são resetadas com a bora de descida de Habilitar assim que possível.</p> 
Saída do eixo	<p>Quando usada no diagrama de blocos de funções, você pode conectar o parâmetro de saída do eixo com o parâmetro de entrada Axis de outro bloco de funções de movimento por conveniência (por exemplo, MC_POWER a MC_HOME). Quando utilizado em um diagrama de lógica ladder, você não pode atribuir uma variável ao parâmetro de saída Axis de outro bloco de funções de movimento porque este é somente-leitura.</p>
Comportamento da saída Done	<p>A saída Done é definida quando a ação comandada é completada com sucesso. Com blocos de funções múltiplos trabalhando no mesmo eixo em uma sequência, a regra a seguir se aplica: Quando um movimento em um eixo é abortado com outro movimento no mesmo eixo sem ter atingido objetivo final, a saída Done não será definida no primeiro bloco de funções.</p> 

Regras gerais para o bloco de funções de movimento (Continuação)

Parâmetro	Regras gerais
Comportamento da saída Busy	<p>Todo bloco de funções possui uma saída Busy, indicando que o bloco de funções ainda não está concluído (para blocos de funções com uma entrada Execute), e novos valores de saída estão pendentes (para blocos de funções com entrada Enable). Busy é definida na borda de subida de Execute e resetada quando uma das saídas Done, Aborted, ou Error é definida, ou é definida na borda de subida de Enable e resetada quando uma das saídas Valid ou Error é definida.</p> <p>É recomendado que o bloco de funções continue executando na varredura do programa enquanto Busy for verdadeira, já que as saídas somente serão atualizadas quando a instrução estiver executando. Por exemplo, em diagrama de lógica ladder, se a linha se tornar falsa antes que a instrução termine de executar, a saída Busy permanecerá verdadeira para sempre mesmo que o bloco de funções termine de executar.</p> 
Saída ativa	Na implementação atual, movimentos bufferizados não são suportados. Consequentemente, as saídas Busy e Active possuem o mesmo comportamento.
Comportamento da saída CommandAborted	<p>CommandAborted é definida quando um movimento comandado é abortado por outro comando de movimento. Quando CommandAborted ocorre, outros sinais de saída como InVelocity são resetados.</p> 
Ativar e validar status	<p>A entrada Enable para blocos de funções lidos possui sensibilidade por níveis. Em cada varredura de programa com a entrada Enable como verdadeira, o bloco de funções realizará uma leitura e atualizará suas saídas. O parâmetro de saída Valid mostra que um conjunto válido de saídas está disponível.</p> <p>A saída Valid é verdadeira enquanto valores de saída válidos estiverem disponíveis e a entrada Enable for verdadeira. Os valores de saída relevantes serão atualizados enquanto a entrada Enable for verdadeira.</p> <p>Se existir um erro de bloco de funções, e os valores de saída relevantes não forem válidos, então a saída válida é definida como falsa. Quando uma condição de erro não existir mais, os valores serão atualizados e a saída Valid será definida novamente.</p>

Regras gerais para o bloco de funções de movimento (Continuação)

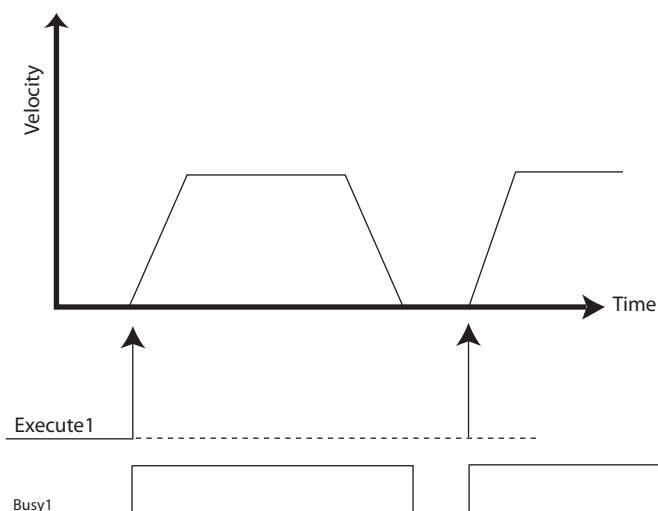
Parâmetro	Regras gerais
Movimento relativo versus movimento absoluto	Movimento relativo não requer que o eixo seja colocando na posição inicial. Ele se refere simplesmente a um movimento em uma direção e distância especificadas. Movimento absoluto requer que o eixo seja colocando na posição inicial. É um movimento a uma posição conhecida dentro do sistema de coordenadas, independentemente de distância e direção. Posição pode ser um valor positivo ou negativo.
Modo com buffer	Para todos os blocos de funções de controle de movimento, o parâmetro de entrada BufferMode é ignorado. Somente movimentos abortados são suportados para essa versão.
Gerenciamento de erros	Todos os blocos possuem duas saídas que gerenciam erros que podem ocorrer durante a execução. Essas saídas são definidas como descrito a seguir: <ul style="list-style-type: none"> • Error – A borda de subida de "Error" informa que um erro ocorreu durante a execução do bloco de funções, onde o bloco de funções não foi concluído com sucesso. • ErrorID – Número do erro. • Tipos de erros: <ul style="list-style-type: none"> • Lógica de bloco de funções (como parâmetros fora da faixa, tentativa de violação de máquina de estados) • Limites ou limites programados atingidos • falha no inversor (Drive Ready é falsa) Para obter mais informações sobre o erro do bloco de funções, consulte Bloco de funções de movimento e ID de erro de status de eixo na página 167 .

Execução simultânea de dois blocos de funções de movimento (saída Busy = Verdadeira)

A regra geral é que quando um bloco de funções de movimento está ocupado, então um bloco de funções **com a mesma instância** (por exemplo, MC_MoveRelative2) não pode ser executado novamente até que o status do bloco de funções não esteja mais ocupado.

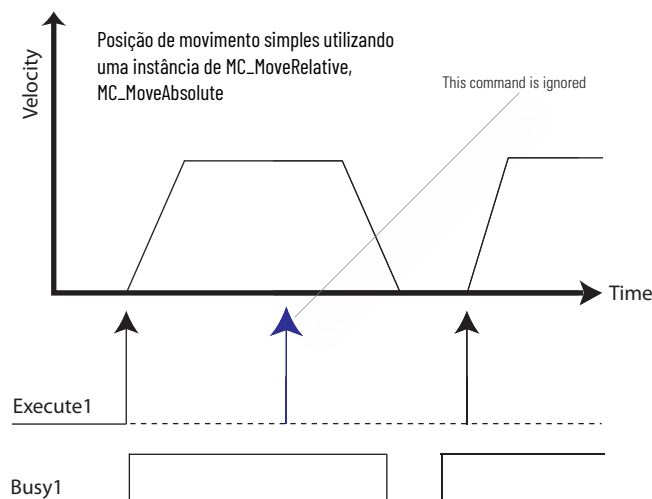


MC_MoveRelative, MC_MoveAbsolute ficarão ocupados até que a posição final seja atingida. MC_MoveVelocity, MC_Halt e MC_Stop ficarão ocupados até que a velocidade final seja atingida.



Quando um bloco de funções de movimento está ocupado, um bloco de funções **com uma instância diferente** (por exemplo, MC_MoveRelative1 e MC_MoveAbsolute1 no mesmo eixo) pode abortar o bloco de funções executando no momento. Isso é muito útil para ajustes dinâmicos à posição, velocidade ou ao parar após uma distância específica.

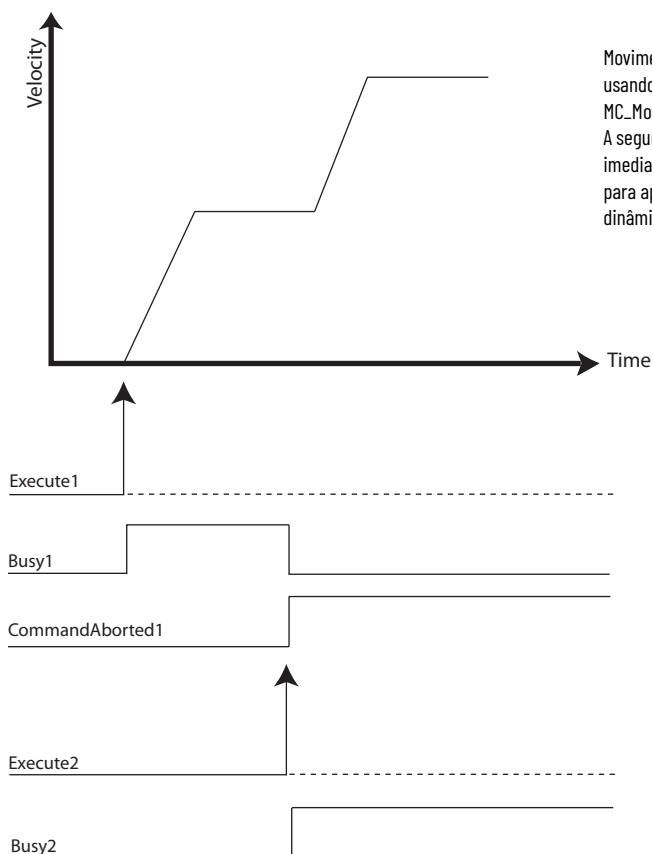
Exemplo: Mover para posição ignorado devido a status Busy



Para movimentos simples, o bloco de funções de movimento é encerrado. Saída Busy indica que o bloco de funções está sendo executado e é necessário permitir que seja concluído antes que a entrada Execute seja alternada novamente.

Se Execute for alternada novamente antes que Busy seja falsa, o novo comando é ignorado. Nenhum erro é gerado.

Exemplo: movimento abortado com sucesso

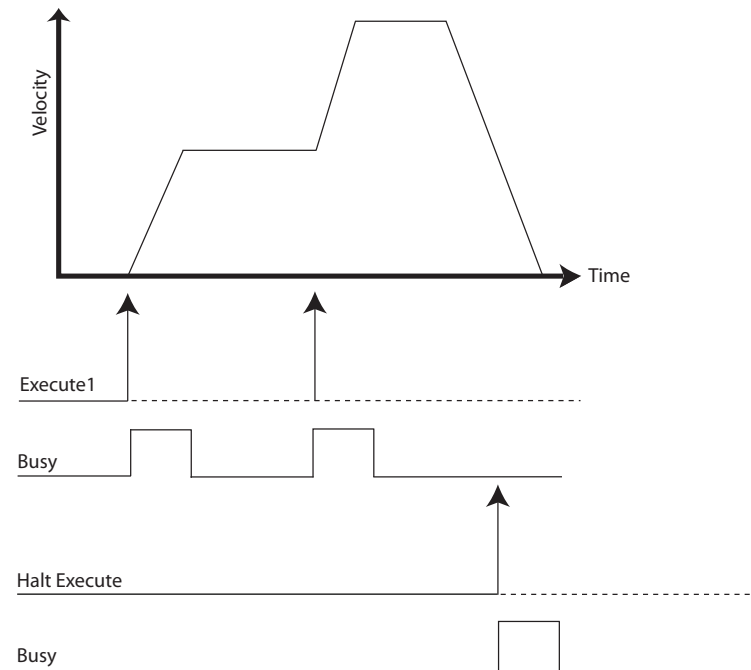


Movimento abortado é possível se estiver usando duas instâncias de MC_MoveRelative, MC_MoveAbsolute. A segunda instância pode abortar imediatamente a primeira (e vice-versa) para aplicações em que correções dinâmicas são necessárias.

Exemplo: Mudar velocidade sem abortar

Ao mudar a velocidade, em geral, um movimento abortado não é necessário já que o bloco de funções fica em status Busy somente durante a aceleração (ou desaceleração). Somente uma única instância do bloco de funções é necessário.

Para causar uma parada no eixo, utilize MC_Halt.

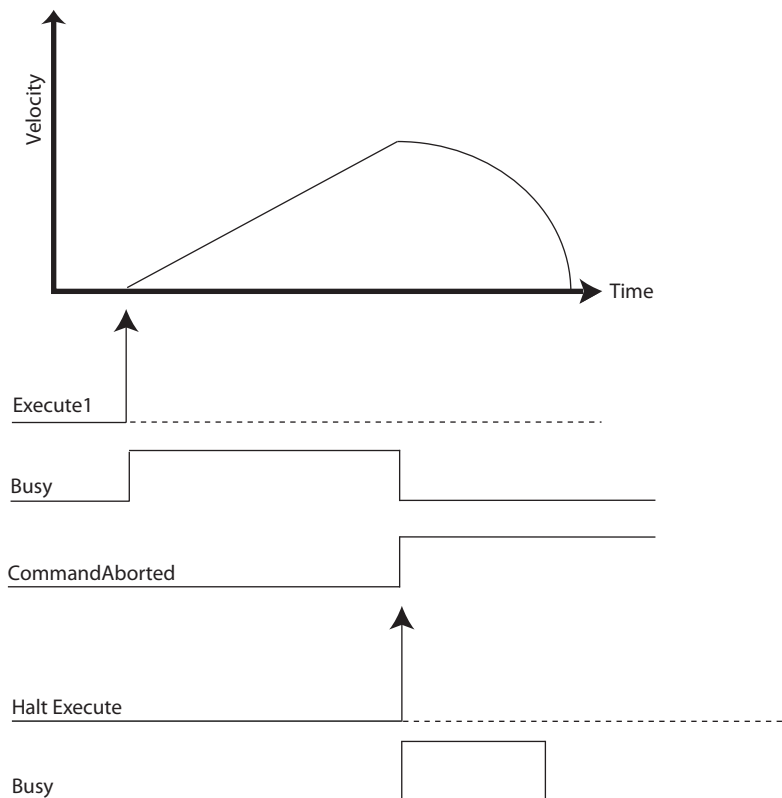


É possível que os blocos de funções de movimento e MC_Halt abortem outro bloco de funções de movimento durante aceleração/desaceleração. Isso não é recomendado, já que o perfil do movimento resultante pode não ser consistente.



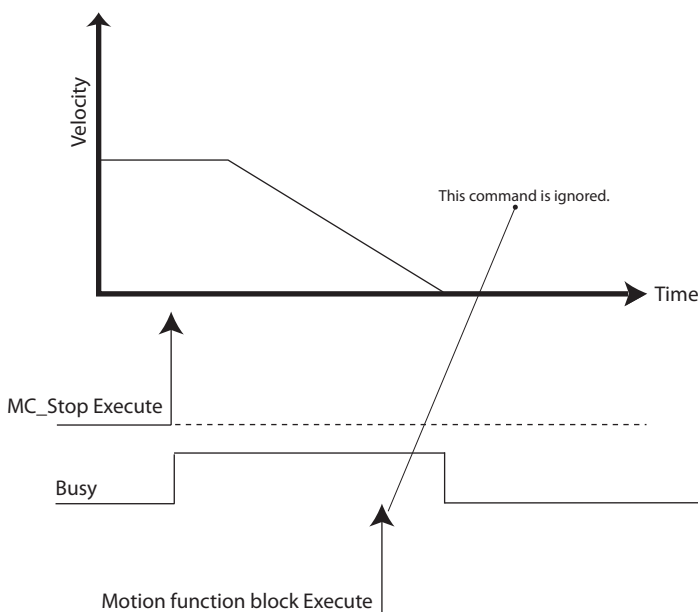
ATENÇÃO: Se MC_Halt aborta outro bloco de funções de movimento durante a aceleração e o parâmetro da entrada Jerk MC_Halt é menor que a Jerk do bloco de funções sendo executado atualmente, a Jerk do bloco de funções sendo executado atualmente é utilizada para prevenir desaceleração excessivamente longa.

Exemplo: Bloco de funções de movimento abortado durante aceleração/desaceleração



IMPORTANTE Se MC_Halt aborta outro bloco de funções de movimento durante a aceleração e o parâmetro da entrada Jerk MC_Halt é menor que a Jerk do bloco de funções sendo executado atualmente, a Jerk do bloco de funções sendo executado atualmente é utilizada para prevenir uma desaceleração excessivamente longa.

Exemplo: Parada de erro utilizando MC_Stop não pode ser abortada



MC_Halt e MC_Stop são ambos utilizados para trazer um eixo a uma parada completa mas MC_Stop é utilizado quando uma situação anormal ocorre.



MC_Stop pode abortar outros blocos de função de posicionamento mas nunca pode ser abortado.



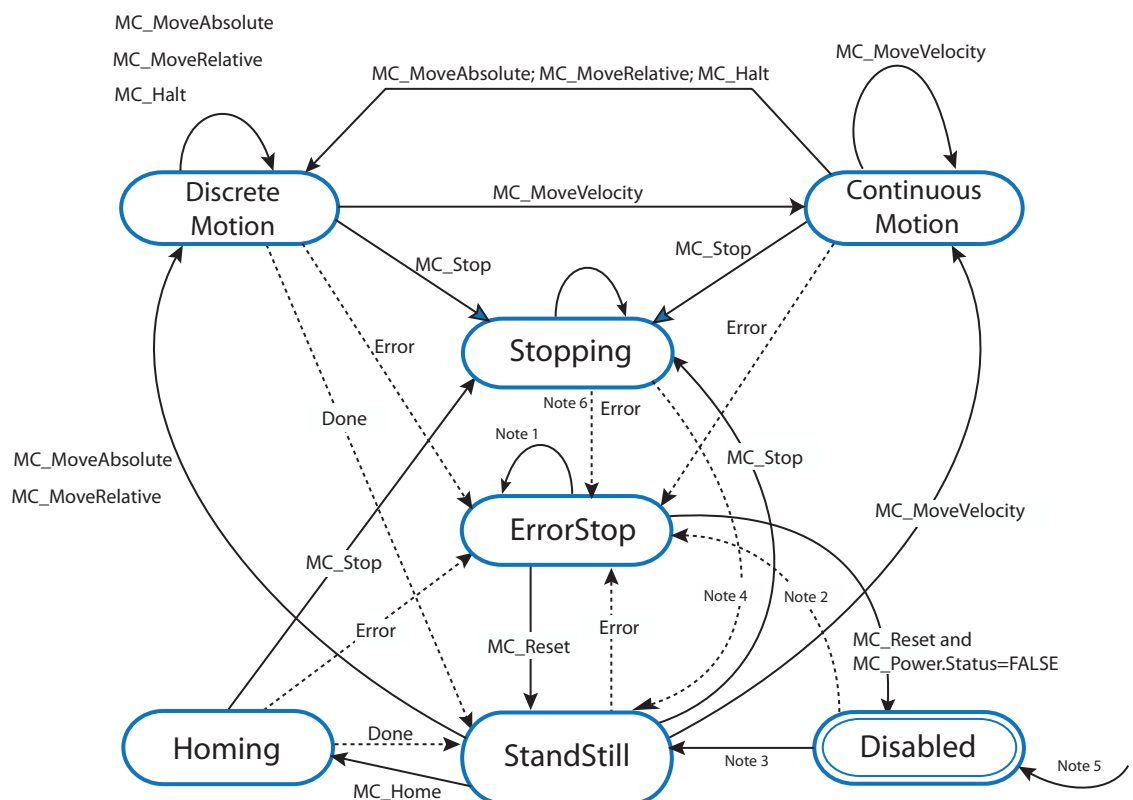
MC_Stop vai para o estado Stopping e a operação normal não pode ser retomada.

Movimento do eixo e parâmetros

O diagrama de estado a seguir ilustra o comportamento do eixo a um nível elevado quando múltiplos blocos de funções de controle de movimento são ativados. A regra básica é que comandos de movimento são sempre recebidos sequencialmente, mesmo se o controlador possuir recursos de processamento paralelo real. Esses comandos agem no diagrama de estado do eixo.

O eixo está sempre em um dos estados definidos, consulte a [Figura 11 na página 158](#). Qualquer comando de movimento é uma transição que muda o estado do eixo e, como consequência, modifica o modo com que o movimento atual é computado.

Figura 11 - Diagrama de estado do eixo de movimento



Observações:

1. Nos estados ErrorStop e Stopping, todos os blocos de funções (exceto MC_Reset) podem ser chamados, apesar de que não serão executados. MC_Reset gera uma transição para o estado StandStill. Se um erro ocorre enquanto o estado máquina está no estado Stopping, é gerada uma transição ao estado ErrorStop. A posição do eixo é atualizada mesmo se o estado do eixo for de parada por erro. Também, o bloco de funções MC_TouchProbe ainda está ativo se ele foi executado antes do estado ErrorStop.
2. Power.Enable = TRUE e existe um erro no eixo.
3. Power.Enable = TRUE e não existe erro no eixo.
4. MC_Stop.Done E NÃO MC_Stop.Execute.
5. Quando MC_Power é chamado com Enable = falso, o eixo vai para o estado Disabled para todos os estados incluindo ErrorStop.
6. Se um erro ocorre enquanto o estado máquina está em estado Stopping, é gerada uma transição ao estado ErrorStop.

Estados de eixo

O estado de eixo pode ser determinado a partir de um dos estados predefinidos a seguir. O estado do eixo pode ser monitorado por meio do recurso monitor de eixo do software Connected Components Workbench no modo de depuração.

Estados de movimento

Valor do estado	Nome do estado
0x00	Desabilitado
0x01	Standstill
0x02	Discrete Motion
0x03	Continuous Motion
0x04	Retorno à posição inicial
0x06	Parando
0x07	Stop Error

Atualização do estado do eixo

Na execução do movimento, apesar do perfil de movimento ser controlado pelo motor de movimento como uma tarefa de plano de fundo, que é independente da varredura POU, a atualização do estado do eixo ainda depende de quando o bloco de funções de movimento relevante é chamado pela varredura POU.

Por exemplo, em um eixo móvel em um POU de lógica ladder (estado de uma linha=verdadeiro), um bloco de funções MC_MoveRelative na linha passa pela varredura e o eixo começa a se mover. Antes que MC_MoveRelative seja concluído, o estado da linha torna-se Falso, e MC_MoveRelative não sofre mais varredura. Nesse caso, o estado desse eixo não pode alternar-se de Discrete Motion para StandStill, mesmo após o eixo parar totalmente, e a velocidade chegar a 0.

The screenshot displays the Rockwell Automation Connected Components Workbench interface. On the left, a ladder logic diagram is shown with a blue 'Always_on' contact connected to the 'EN' (Enable) input of an 'MC_MoveRelative' block. The block's parameters are configured as follows: 'AxisIn' is set to 'Axis0', 'Execute' is set to 'TON_1Q' (which is 'True'), 'Distance' is '2345.66', 'Velocity' is '100.0', 'Acceleration' is '5.0', 'Deceleration' is '5.0', 'Jerk' is '0.0', and 'BufferMode' is '0'. The block's outputs on the right include 'Axis' (0), 'Done' (False), 'Busy' (True), 'Active' (True), 'Commandaborted' (False), 'Error' (False), and 'ErrorID' (0). On the right side of the interface, the 'Variable Monitoring' window is open, showing a tree view with 'Axis0' expanded. The 'Logical Value' column shows the following values: 'Axis0.ErrorFlag' (False), 'Axis0.AxisHomed' (False), 'Axis0.ConstVel' (False), 'Axis0.AccelFlag' (False), 'Axis0.DecelFlag' (False), 'Axis0.AxisState' (2), 'Axis0.ErrorID' (0), 'Axis0.ExtraData' (0), 'Axis0.TargetPos' (2355.66), 'Axis0.CommandPos' (2355.66), 'Axis0.TargetVel' (100.0), and 'Axis0.CommandVel' (0.0). Below this, 'Axis1' is also listed with 'axis0_power' (True) and 'axis1_power' (WAIT).

Limites

O parâmetro Limits define um ponto de escala para o eixo, e trabalha em conjunto com o parâmetro Stop para definir uma condição de escala para o eixo no tipo de parada a aplicar quando certos limites configurados são atingidos.

Existem três tipos de limites de posição de movimento.

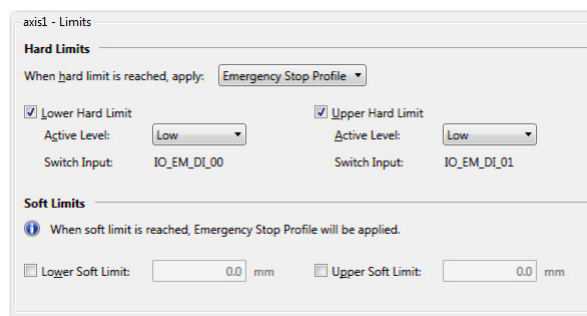
- Limites
- Soft Limits
- Limites de pulso PTO



Consulte [Configuração do eixo de movimento no Connected Components Workbench na página 169](#) para obter informações sobre como configurar limites e perfis de parada e a faixa de valores aceitáveis para cada um.

Se qualquer um desses limites for atingido em um eixo móvel (exceto no retorno ao início), um erro de limite de sobrecurso será relatado e o eixo será parado segundo o comportamento configurado.

Exemplo de configuração de limites no software Connected Components Workbench



Limites físicos

Limites físicos referem-se a sinais de entrada recebidos por dispositivos de hardware físicos como chaves de fim de curso e sensores de proximidade. Esses sinais de entrada detectam a presença da carga nas extensões superior máxima e inferior mínima do movimento permitido da carga ou estrutura móvel que leva a carga, como uma bandeja de carga em um dispositivo de transporte.

Limites de hardware são mapeados para entradas discretas associadas com variáveis/tags de dados.

Quando uma chave de fim de curso é habilitada, o eixo para quando a chave de fim de curso é detectada durante o movimento. Se uma parada por limite com chave de fim de curso é configurada como ATIVADA e o limite é detectado, o movimento é parado imediatamente (ou seja, o pulso PTO é parado imediatamente pelo hardware). Alternativamente, se uma parada por limite com uma chave de fim de curso é configurada como DESATIVADA, o movimento será parado utilizando parâmetros de parada de emergência.

Quando qualquer chave de fim de curso é habilitada, a variável de entrada conectando essa entrada física ainda pode ser utilizada na aplicação do usuário.

Quando uma chave de fim de curso é habilitada, ela será utilizada automaticamente para o bloco de funções MC_Home, se a chave estiver na direção de retorno ao início configurada no software Connected Components Workbench (modo: MC_HOME_ABS_SWITCH or MC_HOME_REF_WITH_ABS). Consulte [Bloco de funções de retorno à posição inicial na página 181](#).

Soft Limits

Limites suaves referem-se a valores de dados que são gerenciados pelo controlador de movimento. Ao contrário de limites de hardware, que detectam a presença de carga física em pontos específicos no movimento permitido da carga, limites suaves são baseados no comando passo a passo e nos parâmetros de carga e de motor.

Limites suaves são exibidos nas unidades definidas pelo usuário. O usuário pode habilitar limites suaves individuais. Para limites suaves não habilitados (independentemente de serem superiores ou inferiores), assume-se um valor infinito.

Limites suaves são ativados somente quando o eixo correspondente é levado à posição inicial. Usuários podem habilitar ou desabilitar os limites suaves, e configurar uma definição de limite superior e inferior por meio do software Connected Components Workbench.

Verificação de limites suaves nos blocos de funções

Blocos de funções	Verificação de limites
MC_MoveAbsolute	A posição desejada será verificada em comparação com os limites suaves antes do início do movimento.
MC_MoveRelative	
MC_MoveVelocity	Os limites suaves serão verificados dinamicamente durante o movimento.

Quando uma chave de fim de curso suave é habilitada, o eixo para quando o limite é detectado durante o movimento. O movimento é parado utilizando parâmetros de parada de emergência.

Se ambos os limites físicos e programados forem configurados como habilitados, para dois limites na mesma direção (superior ou inferior), os limites devem ser configurados de modo que o limite programado seja disparado antes do limite físico.

Limites de pulso PTO

Esse parâmetro de limite não é configurável pelo usuário e é a limitação física do PTO incorporado. Os limites são definidos nos pulsos 0x7FFF0000 e -0x7FFF0000, para limites os inferior e superior, respectivamente.

Limites de pulso PTO são verificados pelo controlador incondicionalmente - ou seja, a verificação está sempre ATIVADA.

Em um movimento descontínuo, para evitar que um eixo móvel entre em um status ErrorStop com limites de pulso PTO de movimento detectados, o usuário precisa prevenir que o valor da posição atual vá além do limite de pulso PTO.

Em um movimento contínuo (movido pelo bloco de funções MC_MoveVelocity), quando o valor da posição atual fica abaixo do limite de pulso PTO, a posição atual de pulso PTO rolará automaticamente para o (ou o limite programado oposto, se estiver ativado), e o movimento contínuo prossegue.

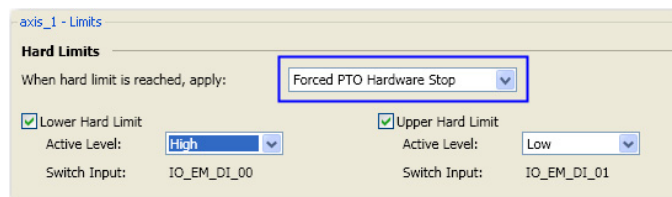
Para um movimento contínuo, se o eixo está na posição inicial e o limite programado na direção do movimento está habilitado, o limite programado será detectado antes que o limite do pulso PTO seja detectado.

Parada de movimento

Existem três tipos de paradas que podem ser configuradas para um eixo.

Parada de hardware imediata

Esse tipo de parada imediata é controlado pelo hardware. Se uma parada por limite com uma chave de fim de curso for habilitada, e o limite tiver sido atingido, o pulso PTO para o eixo será cortado fora imediatamente pelo controlador. A resposta de parada não possui atraso (menos de 1 μ s).



Parada suave imediata

O atraso de resposta máximo possível para esse tipo de parada pode ser tanto quanto o intervalo de tempo de execução do movimento. Esse tipo de parada é aplicável nos seguintes cenários:

- Durante o movimento, quando o limite de pulso PTO do eixo é atingido;
- Um limite é habilitado para um eixo, mas parada por limite com chave de fim de curso está configurada como Desativada. Se a parada de emergência é configurada como parada de software imediata, durante o movimento, quando a chave de fim de curso é detectada;
- Um limite programado é habilitado para um eixo e o eixo foi levado de volta à posição inicial. Se a parada de emergência é configurada como parada suave imediata, durante o movimento, quando detecta-se que o limite programado foi alcançado;
- A parada de emergência é configurada como parada suave imediata. Durante o movimento, o bloco de funções MC_Stop é executado com o parâmetro Deceleration igual a 0.

Parada suave por desaceleração

Parada suave por desaceleração pode ser atrasada tanto quanto o intervalo de tempo de execução do motor de movimento. Esse tipo de parada é aplicada nos seguintes cenários:

- Um limite é habilitado para um eixo, mas parada por limite com chave de fim de curso está configurada como Desativada. Se a parada de emergência é configurada como parada por desaceleração, durante o movimento, quando a chave de fim de curso é detectada;
- Um limite programado é habilitado para um eixo e o eixo foi levado de volta à posição inicial. Se a parada de emergência é configurada como parada por desaceleração, durante o movimento, quando detecta-se por firmware que o limite programado foi alcançado;
- A parada de emergência é configurada como parada por desaceleração. Durante o movimento, o bloco de funções MC_Stop é executado com o parâmetro Deceleration igual a 0.
- Durante o movimento, o bloco de funções MC_Stop é executado com o parâmetro Deceleration não definido como 0.

Direção do movimento

Para movimento de distância (posição), com a posição desejada definida (absoluta ou relativa), a entrada de direção é ignorada.

Para movimento de velocidade, valor de entrada de direção pode ser positivo (1), atual (0) ou negativo (-1). Para qualquer outro valor, somente o sinal (independentemente de positivo ou negativo) é considerado e define se a direção é positiva ou negativa. Isso significa que se o produto de velocidade e direção for -3, então o tipo de direção será negativo.

Tipos de direção suportados por MC_MoveVelocity

Tipo de direção	Valor utilizado ⁽¹⁾	Descrição da direção
Direção positiva	1	Específica para direção de rotação/movimento. Também chamada de direção em sentido horário para movimento de rotação.
Direção atual	0	Direção atual instrui o eixo para continuar o seu movimento com novos parâmetros de entrada, sem mudança de direção. O tipo de direção é válido somente quando o eixo está se movendo e MC_MoveVelocity é chamado.
Direção negativa	-1	Específica para direção de rotação/movimento. Também chamada de direção em sentido anti-horário para movimento de rotação.

(1) Tipo de dados: número inteiro simples.

Elementos de eixo e tipos de dados

Tipo de dados Axis_Ref

A Axis_Ref é uma estrutura de dados que contém informações sobre um eixo de movimentação. Ela é utilizada como uma variável de entrada e saída em todos os blocos de funções de movimento. Uma instância axis_ref é criada automaticamente no software Connected Components Workbench quando o usuário adiciona um eixo de movimento à configuração.

O usuário pode monitorar sua variável em modo de depuração do controlador por meio do software quando o motor de movimento estiver ativo, ou na aplicação de usuário como parte da lógica do usuário. Também podem ser monitorados remotamente por meio de diversos canais de comunicação.

Elementos de dados para Axis_Ref

Nome do elemento	Tipo de dados	Descrição
Axis_ID	UINT8	A ID lógica do eixo atribuída automaticamente pelo software Connected Components Workbench. Este parâmetro não pode ser editado ou visualizado pelo usuário.
ErrorFlag	UINT8	Indica se um erro está ou não presente no eixo.
AxisHomed	UINT8	Indica se a operação de retorno à posição inicial é ou não executada com sucesso para o eixo. Quando o usuário tenta realizar novamente o retorno à posição inicial para um eixo com AxisHomed já definido (retorno à posição inicial realizado com sucesso), e o resultado não for bem-sucedido, o status AxisHomed será removido.
ConsVelFlag	UINT8	Indica se o eixo está ou não em movimento com velocidade constante. Eixo estacionário não é considerado como estando em velocidade constante.
AccFlag	UINT8	Indica se o eixo está ou não em movimento de aceleração ou não.
DecFlag	UINT8	Indica se o eixo está ou não em movimento de desaceleração ou não.
AxisState	UINT8	Indica o estado atual do eixo. Para mais informações, consulte Estados de eixo na página 159 .
ErrorID	UINT16	Indica a causa para o erro do eixo quando esse erro é indicado por ErrorFlag. Esse erro geralmente é resultado de uma falha na execução de um bloco de funções de movimento. Consulte Bloco de funções de movimento e ID de erro de status de eixo na página 167 .
ExtraData	UINT16	Reservado.
TargetPos	REAL (flutuante) ⁽¹⁾	Indica a posição final desejada do eixo para os blocos de funções MoveAbsolute e MoveRelative. Para os blocos de funções MoveVelocity, Stop, e Halt, TargetPos é 0, exceto quando não é limpa a TargetPos definida por blocos de funções de posição anteriores.
CommandPos	REAL (flutuante) ⁽¹⁾	Em um eixo móvel, essa é a posição atual para que o controlador comanda o eixo a mover-se.
TargetVel	REAL (flutuante) ⁽¹⁾	A velocidade máxima desejada atribuída ao eixo por um bloco de funções de movimento. O valor de TargetVel é o mesmo que a configuração de velocidade no bloco de funções atual, ou menor, dependendo de outros parâmetros no mesmo bloco de funções. Esse elemento é um valor com sinal indicando informação relativa à direção. Consulte Precisão de pulso PTO na página 179 para obter mais informações.
CommandVel	REAL (flutuante) ⁽¹⁾	Durante o movimento, este elemento se refere à velocidade que o controlador comanda o eixo a utilizar. Esse elemento é um valor com sinal indicando informação relativa à direção.

(1) Consulte [Resolução de dados reais na página 177](#) para mais informações sobre conversão e arredondamento de dados REAIS.

-
- | | |
|-------------------|--|
| IMPORTANTE | <ul style="list-style-type: none">• Uma vez que um eixo é identificado com erro, e a ID do erro não é zero, o usuário precisa resetar o eixo (utilizando MC_Reset) antes de executar qualquer outro bloco de funções de movimento.• A atualização para status do eixo é realizada no final de um ciclo de varredura do programa, e a atualização é alinhada com a atualização do status do eixo de movimento. |
|-------------------|--|
-

Possibilidades de erro no eixo

Na maioria dos casos, quando uma instrução de bloco de funções de movimento atribuída a um eixo resulta em um erro de bloco de funções, o eixo também é geralmente marcado como estando em erro. O elemento ErrorID correspondente é definido no dado axis_ref para o eixo. Porém, existem situações de exceção em que um eixo não é marcado. A exceção pode ser mas não se limita às seguintes possibilidades:

- Um bloco de funções de movimento instrui um eixo, mas o eixo está em um estado em que o bloco de funções não pode ser executado apropriadamente. Por exemplo, o eixo não possui alimentação, ou está em sequência de retorno à posição inicial, ou em estado de parada de erro.
- Um bloco de funções de movimento instrui um eixo, mas o eixo ainda é controlado por outro bloco de funções de movimento. O eixo não pode permitir que o movimento seja controlado pelo novo bloco de funções sem chegar a uma parada completa. Por exemplo, o novo bloco de funções comanda o eixo a mudar a direção do movimento.
- Quando um bloco de funções de movimento tenta controlar um eixo, mas o eixo ainda é controlado por outro bloco de funções de movimento, e o perfil de movimento recém-definido não pode ser criado pelo controlador. Por exemplo, a aplicação do usuário executa um bloco de funções MC_MoveAbsolute de curva S a um eixo com uma distância curta demais dada quando o eixo está em movimento.
- Quando um bloco de funções de movimento é executado para um eixo, e o eixo está na sequência Stopping ou Error Stopping.

Para as exceções acima, ainda é possível para a aplicação do usuário executar um bloco de funções de movimento bem-sucedido para o eixo após o estado do eixo mudar.

Tipo de dado MC_Engine_Diag

O tipo de dado MC_Engine_Diag contém informações de diagnóstico sobre o motor de movimento incorporado. Ele pode ser monitorado em modo de depuração por meio do software Connected Components Workbench quando o motor de movimento estiver ativo, ou por meio da aplicação de usuário como parte da lógica do usuário. Também podem ser monitorados remotamente por meio de diversos canais de comunicação.

Uma instância MC_Engine_Diag é criada automaticamente no software Connected Components Workbench quando o usuário adiciona o primeiro eixo de movimento à configuração de movimento. Essa instância é compartilhada por todos os eixos de movimento configurados pelo usuário.

Elementos de dados para MC_Engine_Diag

Nome do elemento	Tipo de dados
MCEngState	UINT16
CurrScantime ⁽¹⁾	UINT16
MaxScantime ⁽¹⁾	UINT16
CurrEngineInterval ⁽¹⁾	UINT16
MaxEngineInterval ⁽¹⁾	UINT16
ExtraData	UINT16

(1) A unidade de tempo para esse elemento é o microssegundo. Essa informação de diagnóstico pode ser utilizada para otimizar a configuração de movimento o ajuste da lógica de aplicação do usuário.

Estados MCEngstate

Nome do estado	Estado	Descrição
MCEng_Idle	0x01	Existe o MC engine (ao menos um eixo definido), mas o motor está inativo já que nenhum eixo está se movendo. Os dados de diagnóstico do motor não estão sendo atualizados.
MCEng_Running	0x02	Existe o MC engine (ao menos um eixo definido) e o motor está funcionando. Os dados de diagnóstico estão sendo atualizados.
MCEng_Faulted	0x03	Motor MC existe, mas o motor está com falha.

Códigos de erro de bloco de funções e status de eixo

Todos os blocos de funções de controle de movimento compartilham a mesma definição de ErrorID.

[Tabela 14 na página 167.](#)



Código de erro 128 é informação de advertência para indicar que o perfil de perfil de movimento foi alterado e que a velocidade foi ajustada para um valor menor mas o bloco de funções pode ser executado com sucesso.

Tabela 14 - Bloco de funções de movimento e ID de erro de status de eixo

ID do erro	MACRO da ID do erro	Descrição do erro para bloco de funções	Descrição do erro para status do eixo ⁽¹⁾
00	MC_FB_ERR_NO	Execução do bloco de funções bem-sucedida.	O eixo está em estado operacional.
01	MC_FB_ERR_WRONG_STATE	O bloco de funções não pode ser executado porque o eixo não está no estado correto. Verifique o estado do eixo.	O eixo não está operacional devido a detecção de estado de eixo incorreto durante a execução de um bloco de funções. Restaure o estado do eixo utilizando o bloco de funções MC_Reset.
02	MC_FB_ERR_RANGE	O bloco de funções não pode ser executado porque não há parâmetro(s) dinâmico(s) inválido(s) (velocidade, aceleração, desaceleração ou variação da aceleração) definidos no bloco de funções. Corrija a configuração para os parâmetros dinâmicos no bloco de funções baseando-se na página de configuração Axis Dynamics.	O eixo não está operacional devido a parâmetro(s) dinâmico(s) de eixo inválido(s) (velocidade, aceleração, desaceleração ou variação da aceleração) definidos em um bloco de funções. Restaure o estado do eixo utilizando o bloco de funções MC_Reset. Corrija a configuração para os parâmetros dinâmicos no bloco de funções baseando-se na página de configuração Axis Dynamics.
03	MC_FB_ERR_PARAM).	O bloco de funções não pode ser executado porque não há parâmetros inválidos diferentes de velocidade, aceleração, desaceleração ou variação da aceleração definidos no bloco de funções. Corrija a configuração para os parâmetros (por exemplo, modo ou posição) para o bloco de funções.	O eixo não está operacional devido a parâmetro(s) inválido(s) diferentes de velocidade, aceleração, desaceleração ou variação da aceleração definidos em um bloco de funções. Restaure o estado do eixo utilizando o bloco de funções MC_Reset. Corrija a configuração para os parâmetros (por exemplo, modo ou posição) para o bloco de funções.
04	MC_FB_ERR_AXISNUM	O bloco de funções não pode ser executado porque o eixo não existe, os dados de configuração do eixo estão corrompidos, ou o eixo não está configurado corretamente.	Falha interna de movimento, ID do erro = 0x04. Contate o suporte técnico.
05	MC_FB_ERR_MECHAN	O bloco de funções não pode ser executado porque o eixo está com falha devido a problemas mecânicos ou no inversor. Verifique a conexão entre o inversor e o controlador (sinais Drive Ready e In-Position), e certifique-se de que o inversor esteja operando normalmente.	O eixo não está operacional devido a problemas mecânicos ou de inversor. Verifique a conexão entre o inversor e o controlador (sinais Drive Ready e In-Position), e certifique-se de que o inversor esteja operando normalmente. Restaure o estado do eixo utilizando o bloco de funções MC_Reset.
06	MC_FB_ERR_NOPOWER	O bloco de funções não pode ser executado porque o eixo não está ligado à alimentação elétrica. Ligue a alimentação ao eixo utilizando o bloco de funções MC_Power.	O eixo não está ligado à alimentação elétrica. Ligue a alimentação ao eixo utilizando o bloco de funções MC_Power. Restaure o estado do eixo utilizando o bloco de funções MC_Reset.
07	MC_FB_ERR_RESOURCE	O bloco de funções não pode ser executado porque o recurso exigido pelo bloco de funções é controlado por algum outro bloco de funções ou está indisponível. Certifique-se de que o recurso exigido pelo bloco de funções esteja disponível para uso. Alguns exemplos: <ul style="list-style-type: none"> • O bloco de funções MC_power tenta controlar o mesmo eixo. • O bloco de funções MC_Stop é executado para o mesmo eixo simultaneamente. • Dois ou mais blocos de funções MC_TouchProbe são executados para o mesmo eixo simultaneamente. 	O eixo não está operacional devido ao recurso exigido por um bloco de funções estar sob controle de outro bloco de funções, ou indisponível. Certifique-se de que o recurso exigido pelo bloco de funções esteja disponível para uso. Restaure o estado do eixo utilizando o bloco de funções MC_Reset.
08	MC_FB_ERR_PROFILE	O bloco de funções não pode ser executado porque o perfil de movimento definido no bloco de funções não pode ser atingido. Corrija o perfil no bloco de funções.	O eixo não está operacional devido a um perfil de movimento definido em um bloco de funções não poder ser atingido. Restaure o estado do eixo utilizando o bloco de funções MC_Reset. Corrija o perfil no bloco de funções.

Tabela 14 - Bloco de funções de movimento e ID de erro de status de eixo (Continuação)

ID do erro	MACRO da ID do erro	Descrição do erro para bloco de funções	Descrição do erro para status do eixo ⁽¹⁾
09	MC_FB_ERR_VELOCITY	<p>O bloco de funções não pode ser executado porque o perfil de movimento solicitado no bloco de funções não pode ser atingido devido à velocidade atual do eixo.</p> <p>Alguns exemplos:</p> <ul style="list-style-type: none"> O bloco de funções solicita ao eixo que reverta a direção enquanto o eixo está em movimento. O perfil de movimento exigido não pode ser atingido devido à velocidade atual ser alta ou baixa demais. <p>Verifique a configuração do perfil de movimento no bloco de funções, e corrija o perfil, ou execute novamente o bloco de funções quando a velocidade do eixo for compatível com o perfil de movimento solicitado.</p>	<p>O eixo não está operacional. O perfil de movimento solicitado no bloco de funções não pode ser atingido por causa da velocidade atual do eixo.</p> <p>Alguns exemplos:</p> <ul style="list-style-type: none"> O bloco de funções solicita ao eixo que reverta a direção enquanto o eixo está em movimento. O perfil de movimento exigido não pode ser atingido devido à velocidade atual ser alta ou baixa demais. <p>Restaurar o estado do eixo utilizando o bloco de funções MC_Reset.</p> <p>Corrija o perfil de movimento no bloco de funções, ou execute novamente o bloco de funções quando a velocidade do eixo for compatível com o perfil de movimento solicitado.</p>
10	MC_FB_ERR_SOFT_LIMIT	<p>Esse bloco de funções não pode ser executado, visto que terminará movendo-se além do limite programado, ou o bloco de funções será abortado conforme o limite programado tiver sido atingido.</p> <p>Verifique as configurações de velocidade ou posição desejada no bloco de funções, ou ajuste a configuração de limite programado.</p>	<p>O eixo não está operacional devido à detecção de um erro de limite programado, ou devido a um erro de limite programado esperado em um bloco de funções.</p> <p>Restaurar o estado do eixo utilizando o bloco de funções MC_Reset.</p> <p>Verifique as configurações de velocidade ou posição desejada para o bloco de funções, ou ajuste a configuração de limite programado.</p>
11	MC_FB_ERR_HARD_LIMIT	<p>Esse bloco de funções é abortado conforme o estado ativo de chave de fim de curso for detectado seja durante o movimento do eixo, seja antes que esse movimento comece.</p> <p>Mova o eixo para longe da chave de fim de curso na direção oposta.</p>	<p>O eixo não está operacional devido a um erro de limite detectado.</p> <p>Restaurar o estado do eixo utilizando o bloco de funções MC_Reset, e então mova o eixo para longe da chave de fim de curso na direção oposta.</p>
12	MC_FB_ERR_LOG_LIMIT	<p>Esse bloco de funções não pode ser executado, visto que terminará movendo-se além do limite lógico do acumulador PTO, ou o bloco de funções será abortado conforme o limite lógico do acumulador PTO tiver sido atingido.</p> <p>Verifique as configurações de velocidade ou posição desejada para o bloco de funções. Ou ainda, utilize o bloco de funções MC_SetPosition para ajustar o sistema de coordenadas de eixo.</p>	<p>O eixo não está operacional devido à detecção de um erro de limite lógico do acumulador PTO, ou devido a um erro de limite lógico do acumulador PTO esperado em um bloco de funções.</p> <p>Restaurar o estado do eixo utilizando o bloco de funções MC_Reset.</p> <p>Verifique as configurações de velocidade ou posição desejada para o bloco de funções. Ou ainda, utilize o bloco de funções MC_SetPosition para ajustar o sistema de coordenadas de eixo.</p>
13	MC_FB_ERR_ENGINE	<p>Um erro de execução de motor de movimento é detectado durante a execução deste bloco de funções.</p> <p>Desligue e ligue a alimentação para toda a configuração de movimento, incluindo controlador, inversores e atuadores, e então faça novamente o download da aplicação do usuário.</p> <p>Se o erro persistir, contate o suporte técnico.</p>	<p>O eixo não está operacional devido a um erro de execução de movimento.</p> <p>Desligue e ligue a alimentação para toda a configuração de movimento, incluindo controlador, inversores e atuadores, e então faça novamente o download da aplicação do usuário.</p> <p>Se a falha persistir, contate o seu representante local de suporte técnico Rockwell Automation. Para as informações de contato, consulte: rok.auto/support</p>
16	MC_FB_ERR_NOT_HOMED	<p>O bloco de funções não pode ser executado porque primeiro o eixo precisa ser levado à posição inicial.</p> <p>Execute retorno à posição inicial para o eixo utilizando o bloco de funções MC_Home.</p>	<p>O eixo não está operacional porque o não foi retornado à posição inicial.</p> <p>Restaurar o estado do eixo utilizando o bloco de funções MC_Reset.</p>
128	MC_FB_PARAM_MODIFIED	<p>Atenção: O parâmetro de movimento exigido para o eixo foi ajustado.</p> <p>O bloco de funções é executado com sucesso.</p>	<p>Falha interna de movimento, ID do erro = 0x80.</p> <p>Contate o seu representante local de suporte técnico Rockwell Automation. Para informações de contato, consulte: rok.auto/support</p>

(1) Você pode visualizar o status do eixo por meio do recurso Axis Monitor do software Connected Components Workbench.

Quando um bloco de funções de controle de movimento termina com um erro e o eixo está no estado ErrorStop, na maioria dos casos, o bloco de funções MC_Reset (ou MC_Power Off/On e MC_Reset) pode ser utilizado para fazer com que o eixo seja recuperado. Com isso, o eixo pode retornar à operação em movimento normal sem parar a operação do controlador.

Gerenciamento de falha grave

Caso o controlador encontre problemas em que a recuperação não seja possível por meio dos blocos de funções Stop, Reset ou Power, a operação do controlador será suspensa e uma falha grave será reportada.

[Tabela 15 na página 169](#) defini os códigos de falhas graves relacionadas a movimento para os controladores Micro830 e Micro850 e Micro870.

Tabela 15 - Códigos de erro e descrição das falhas graves

Valor da falha grave	MACRO ID da falha	Descrição da falha grave
0xF100	EP_MC_CONFIG_GEN_ERR	Existe um erro de configuração geral detectado na configuração de movimento baixada do software Connected Components Workbench, como número do eixo, intervalo de execução do movimento sendo configurado fora da faixa. Quando essa falha grave for relatada, não poderiam existir eixos no estado ErrorStop.
0xF110	EP_MC_RESOURCE_MISSING	Configuração de movimento possui problemas de diferença com o recurso de movimento baixado do controlador. Existem alguns recursos de movimento faltando. Quando essa falha grave for relatada, não poderiam existir eixos no estado ErrorStop.
0xF12x	EP_MC_CONFIG_AXES_ERR	A configuração de movimento para o eixo não pode ser suportada por esse catálogo, ou a configuração possui algum conflito de recurso com algum outro eixo de movimento, que foi configurada mais cedo. A razão possível pode ser velocidade ou aceleração máximas configuradas fora da faixa suportada. x = a ID lógica do eixo (0 a 3).
0xF15x	EP_MC_ENGINE_ERR	Há um erro na lógica do motor de movimento (problema de lógica do firmware ou colisão de memória) para um eixo detectado durante a operação cíclica do motor de movimento. Uma razão possível pode ser um erro de memória/dados no motor de movimento. (Esse é o erro de operação do motor de movimento, e não deve ocorrer em condições normais.) x = a ID lógica do eixo (0 a 3).

Configuração do eixo de movimento no Connected Components Workbench

Um máximo de três eixos de movimentação podem ser configurados por meio do software Connected Components Workbench. Para adicionar, configurar, atualizar, excluir e monitorar um eixo no software Connected Components Workbench, consulte as próximas seções.



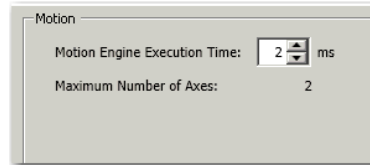
Mudanças de configuração precisam ser compiladas e baixadas pelo controlador para entrarem em efeito.



Valores para os diferentes parâmetros de eixo de movimento são validades segundo um conjunto de relações e faixa absoluta pré-determinada. Consulte [Validação de parâmetros de eixo de movimento na página 180](#) para uma descrição das relações entre os parâmetros.

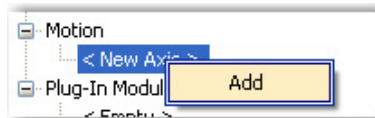
Adicionar um novo eixo

IMPORTANTE Tempo de execução do motor de movimento



Quando um eixo é adicionado à configuração, o tempo de execução do movimento pode ser configurado entre 1 e 10 ms (padrão: 1 ms). Esse parâmetro global se aplica a todas as configurações de eixo de movimento.

1. Na árvore de configuração de dispositivo, clique com o botão direito em <New Axis>. Clique em Add.



2. Forneça um nome de eixo. Clique em Enter.

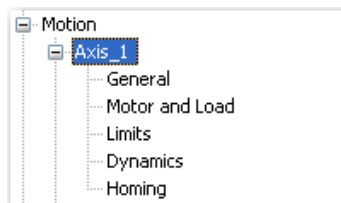


O nome precisa começar com um caractere de letra ou sublinhado, seguido por caracteres de letra ou sublinhado simples.



Você também pode pressionar F2 para editar o nome do eixo.

3. Expanda o eixo recém-criado para ver as categorias de configuração a seguir:
 - Gerais
 - Motor and Load
 - Limites
 - Dinâmica
 - Retorno à posição inicial

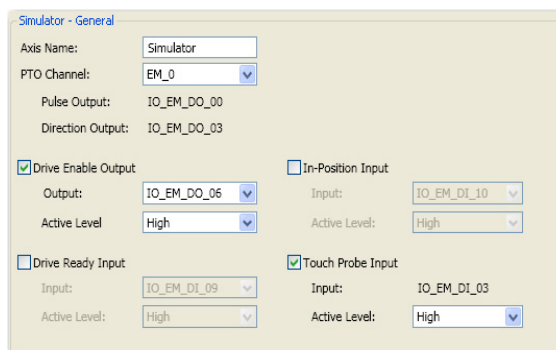


Para ajudá-lo a editar essas propriedades de movimento, consulte [Editar a configuração de eixo na página 171](#). Você também pode aprender mais sobre parâmetros de configuração de eixo.

Editar a configuração de eixo

Parâmetros gerais

1. Na árvore de configuração de eixo, clique em General. A guia Propriedades gerais do <Axis Name> aparece.



2. Editar os parâmetros gerais. Consulte a [Tabela 16 na página 171](#) para uma descrição dos parâmetros gerais de configuração para um eixo de movimento.

IMPORTANTE Para editar esses parâmetros gerais, consulte [Sinais de entrada e saída na página 146](#) para obter mais informações sobre saídas fixas e configuráveis.

Tabela 16 - Parâmetros gerais

Parâmetro	Descrição e valores
Axis Name	Definido pelo usuário. Fornece um nome para o eixo de movimento.
PTO Channel	Mostra a lista de canais PTO disponíveis.
Saída de pulso	Apresenta o nome da variável lógica do canal de saída de direção segundo o valor de canal PTO que foi atribuído.
Direction output	Apresenta o nome da variável lógica do canal de saída de direção segundo o valor de canal PTO que foi atribuído.
Drive Enable Output	Flag de habilitação de saída de servo ativado. Marque a caixa de opção para habilitar.
- Output	A lista de variáveis de saída digital disponíveis que podem ser atribuídas como saída de servo/inversor.
- Active Level	Defina como alto (padrão) ou baixo.
In-position Input	Marque a caixa de opção para habilitar o monitoramento de entrada em posição.
- Input	Lista de variáveis de entrada digital para monitoramento de entrada em posição. Selecione uma entrada.
- Active Level	Defina como alto (padrão) ou baixo.
Entrada de inversor pronto	Flag de habilitação de entrada Servo Ready. Marque a caixa de opção para habilitar a entrada.
- Input	A lista de variáveis de entrada digital. Selecione uma entrada.
- Active Level	Defina como alto (padrão) ou baixo.
Entrada Touch probe	Configure se uma entrada para sensor de toque é utilizada. Marque a caixa de opção para habilitar a entrada de sensor de toque.
- Input	A lista de variáveis de entrada digital. Selecione uma entrada.
- Active Level	Configure o nível ativo para a entrada de sensor de toque como alto (padrão) ou baixo.

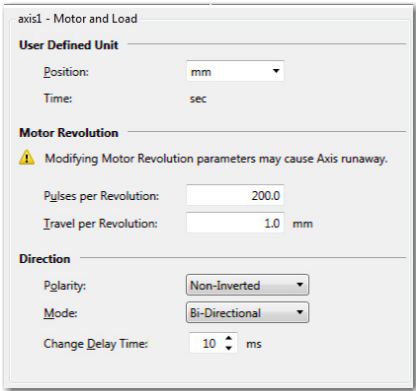
Nomeação de canal PTO

Os nomes dos canais PTO incorporados têm o prefixo EM (incorporado) e cada canal PTO disponível é numerado a partir de 0. Por exemplo, um controlador que suporta três eixos terá os seguintes canais PTO disponíveis:

- EM_0
- EM_1
- EM_2

Motor e Carga

Edite as propriedades de carga do motor conforme definido na [Tabela 17 na página 172](#).



IMPORTANTE Certos parâmetros para Motor e Carga são valores reais. Para obter mais informações, consulte [Resolução de dados reais na página 177](#)

Tabela 17 - Parâmetros de motor e carga

Parâmetro	Descrição e valores
Unidade definida pelo usuário	Define a conversão de escala de unidade definida pelo usuário que corresponde aos valores do seu sistema mecânico. Essas unidades deverão ser levadas adiante em todos os eixos de monitoramento e comando nos valores de unidade definida pelo usuário por funções de programação, configuração e monitoramento.
Posição	Selecione qualquer uma das opções a seguir: <ul style="list-style-type: none">- mm- cm- polegadas- rotações- unidade personalizada (formato ASCII com até 7 caracteres)
Tempo	somente leitura Predefinido em segundos.
Revolução do motor	Define os valores de pulso por revolução e percurso por revolução.
Pulso por revolução ⁽¹⁾	Define o número de pulsos necessários para obter uma revolução do motor de acionamento. Intervalo: 0,0001 a 8.388.607 Padrão: 200,0
Percurso por revolução ⁽¹⁾	Percurso por revolução define a distância, seja linear ou rotativa, pela qual a carga se move por revolução do motor. Intervalo: 0,0001 a 8.388.607. Padrão: 1,0 unidade de usuário.
Direção	Define polaridade, modo, e mudança de valores de tempo de retardo.

Tabela 17 - Parâmetros de motor e carga (Continuação)

Parâmetro	Descrição e valores
Polarity	Polaridade da direção determina se o sinal de direção recebido pelo controlador como uma entrada discreta deveria ser interpretado na entrada conforme recebido pelo controlador de movimento, (ou seja, o caso não invertido), ou se o sinal deve ser invertido antes de ser interpretado pela lógica de controle de movimento. Defina como Invertido ou Não Invertido (padrão).
Modo	Defina como Bidirecional (padrão), direção Positiva (sentido horário) ou Negativa (sentido anti-horário).
Tempo de atraso para mudança	Configure entre 0 a 100 ms. Valor-padrão é 10 ms.

(1) O parâmetro é definido como valor REAL (flutuante) no software Connected Components Workbench. Para aprender mais sobre conversões e arredondamento de valores REAIS, consulte [Resolução de dados reais na página 177](#).



Uma borda vermelha em um campo de entrada indica que um valor inválido foi inserido. Role pelo campo para ver uma mensagem de dica que permitirá que você saiba a faixa de valores válidos para o parâmetro. Forneça o valor válido.



ATENÇÃO: Modificando parâmetros de revolução de motor pode causar perda de controle sobre o eixo.

Limites

Edite os parâmetros Limits segundo a tabela abaixo.



ATENÇÃO: Para aprender mais sobre os diferentes tipos de limites, consulte [Limites na página 160](#).

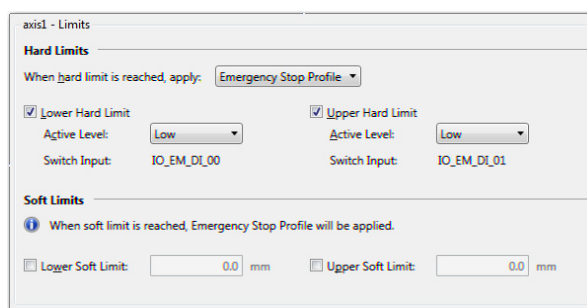


Tabela 18 - Parâmetros de limites

Parâmetro ⁽¹⁾	Valor
Limites	Define limites inferior e superior para o eixo.
Quando os limites forem atingidos, aplique	Configure se deseja ou não realizar uma parada de hardware PTO forçada (desligamento imediato da saída de pulso) ou deseja desacelerar ou não (deixe a saída de pulso ativada e utilize os valores de desaceleração conforme definido no perfil de parada de emergência). Defina como qualquer uma das opções a seguir: <ul style="list-style-type: none"> Forced PTO Hardware Stop Emergency Stop Profile
Lower Hard Limit	Clique na caixa de marcar para habilitar um limite inferior.
Active Level (para limite físico inferior)	High ou Low.
Upper Hard Limit	Clique na caixa de marcar para habilitar.

Tabela 18 - Parâmetros de limites (Continuação)

Parâmetro ⁽¹⁾	Valor
Active Level (para limite superior físico)	High ou Low.
Soft Limits	Define valores dos limites suaves inferior e superior.
Lower Soft Limit ⁽²⁾	limite programado inferior deve ser menor que o limite programado superior. 1. Clique na caixa de marcar para habilitar um limite programado inferior/superior. 2. Especifique um valor (em mm).
Upper Soft Limit ⁽²⁾	

(1) Para converter de unidades definidas pelo usuário para pulsos:

$$\text{Valor em unidade definida pelo usuário} = \text{Valor em pulso} \times \frac{\text{Percurso por revolução}}{\text{Pulso por revolução}}$$

(2) O parâmetro é definido como valor REAL (flutuante) no software Connected Components Workbench. Para aprender mais sobre conversões e arredondamento de valores REAIS, consulte [Resolução de dados reais na página 177](#).



Uma borda vermelha em um campo de entrada indica que um valor inválido foi inserido. Role pelo campo para ver uma mensagem de dica que permitirá que você saiba a faixa de valores válidos para o parâmetro. Forneça o valor válido.

- Clique em Dynamics. <Axis Name> – a guia Dynamics aparece. Edite os parâmetros Dynamics com base nos valores da [Tabela 19 na página 174](#).

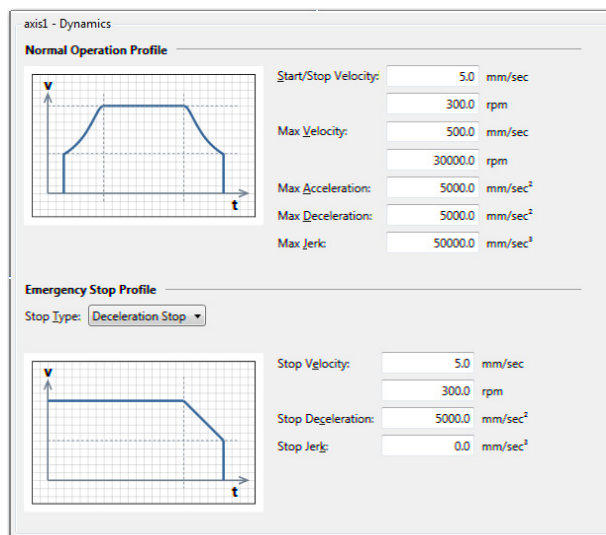


Tabela 19 - Parâmetros dinâmicos

Parâmetro	Valores
Velocidade iniciar/parar ⁽¹⁾⁽²⁾	O intervalo é baseado nos parâmetros Motor e Carga (Consulte Parâmetros de motor e carga na página 172) usando: Intervalo 1 a 10.000.000 pulsos/s Padrão: 300 rpm Por exemplo, você pode configurar o valor entre 0,005 a 500 mm/s para 200 pulsos por rotação de 1 mm. ⁽³⁾ O valor em RPM é preenchido automaticamente quando um valor em unidades definidas pelo usuário é especificado, mas o usuário também pode inserir inicialmente um valor em RPM. Velocidade de partida/parada não deve ser maior que a velocidade máxima.
Start/Stop Velocity em RPM ⁽¹⁾⁽²⁾	
Max Velocity ⁽¹⁾⁽²⁾	

Tabela 19 - Parâmetros dinâmicos (Continuação)

Parâmetro	Valores
Max Acceleration ⁽¹⁾	O intervalo é baseado nos parâmetros Motor e Carga (Consulte Parâmetros de motor e carga na página 172) usando: Intervalo: 1 a 10.000.000 pulso/s ² Padrão: 10.000.000 pulso/s ²
Max Deceleration ⁽¹⁾	O intervalo é baseado nos parâmetros Motor e Carga (Consulte Parâmetros de motor e carga na página 172) usando: Intervalo: 1 a 100.000 pulso/s ² Padrão: 10.000.000 pulso/s ²
Max Jerk ⁽¹⁾	O intervalo é baseado nos parâmetros Motor e Carga (Consulte Parâmetros de motor e carga na página 172) usando: Intervalo: 0 a 10.000.000 pulso/s ³ Padrão: 10.000.000 pulso/s ³
Emergency Stop Profile	Define os valores de tipo de parada, velocidade, desaceleração e variação da aceleração.
Stop Type	Definido como parada por desaceleração (padrão) ou parada imediata.
Stop Velocity ⁽¹⁾	O intervalo é baseado nos parâmetros Motor e Carga (Consulte Parâmetros de motor e carga na página 172) usando: Intervalo: 1 a 100.000 pulso/s Padrão: 300 rpm
Stop Deceleration ⁽¹⁾	O intervalo é baseado nos parâmetros Motor e Carga (Consulte Parâmetros de motor e carga na página 172) usando: Intervalo: 1 a 10.000.000 pulso/s. Padrão: 300,0 rpm ²
Stop Jerk ⁽¹⁾	O intervalo é baseado nos parâmetros Motor e Carga (Consulte Parâmetros de motor e carga na página 172) usando: Intervalo: 0 a 10.000.000 pulso/s ³ Padrão: 0,0 rpm ³ (Desabilitado)

(1) O parâmetro é definido como valor REAL (flutuante) no software Connected Components Workbench. Para aprender mais sobre conversões e arredondamento de valores REAIS, consulte [Resolução de dados reais na página 177](#).

(2) A fórmula para derivação de RPM em unidade definida pelo usuário, e vice-versa:

$$v \text{ (em RPM)} = \frac{v \text{ (em unidade definida pelo usuário/s)} \times 60 \text{ s}}{\text{percurso por revolução (em unidade definida pelo usuário)}}$$

(3) Para converter de valor de parâmetro de pulso para unidades definidas pelo usuário:

$$\text{Valor em unidade definida pelo usuário} = \text{Valor em pulso} \times \frac{\text{Percurso por revolução}}{\text{Pulso por revolução}}$$



Uma borda vermelha em um campo de entrada indica que um valor inválido foi inserido. Role pelo campo para ver uma mensagem de dica que permitirá que você saiba a faixa de valores válidos para o parâmetro. Forneça o valor válido.

- Defina os parâmetros de retorno à posição inicial com base na descrição na [Tabela 20](#). Clique em Homing.

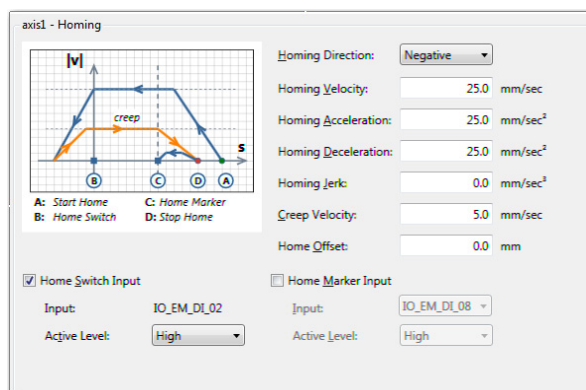


Tabela 20 - Parâmetros de retorno à posição inicial

Parâmetro	Intervalo de valores
Homing Direction	Positivo (sentido horário) ou negativo (sentido anti-horário).
Homing Velocity ⁽¹⁾	Intervalo: 1 a 100.000 pulsos/s Padrão: 5.000,0 pulsos/s (25,0 mm/s) NOTA: Velocidade de retorno à posição inicial não deve ser maior que a velocidade máxima.
Homing Acceleration ⁽¹⁾	Intervalo: 1 a 10.000.000 pulso/s ² Padrão: 5.000,0 pulsos/s ² (25,0 mm/s ²) NOTA: Velocidade de retorno à posição inicial não deve ser maior que a aceleração máxima, Maximum Acceleration.
Homing Deceleration ⁽¹⁾	Intervalo: 1 a 10.000.000 pulso/s ² Padrão: 5.000,0 pulsos/s ² (25,0 mm/s ²) NOTA: Velocidade de retorno à posição inicial não deve ser maior que a desaceleração máxima, Maximum Deceleration.
Homing Jerk ⁽¹⁾	Intervalo: 0 a 10.000.000 pulso/s ³ Padrão: 0,0 pulso/s ³ (0,0 mm/s ³) NOTA: Velocidade de retorno à posição inicial não deve ser maior que a variação de aceleração máxima, Maximum Jerk.
Creep Velocity ⁽¹⁾	Intervalo: 1 a 5.000 pulsos/s Padrão: 1.000,0 pulsos/s (5,0 mm/s) NOTA: Homng Creep Velocity não deve ser maior que a velocidade máxima, Maximum Velocity.
Homing Offset ⁽¹⁾	Intervalo: -1073741824 a +1073741824 pulsos Padrão: 0,0 pulsos/s (0,0 mm)
Home Switch Input	Habilite a entrada da chave de posição inicial clicando na caixa de marcar.
- Input	Valor somente-leitura especificando a variável de entrada para entrada de chave de posição inicial.
- Active Level	Alto (padrão) ou baixo.
Home Marker Input	Habilite a configuração de uma variável de entrada digital clicando na caixa de marcar.
- Input	Especifique a variável de entrada digital para entrada de marcador de posição inicial.
- Active Level	Configure o nível ativo para a entrada de chave de posição inicial como alto (padrão) ou baixo.

(1) O parâmetro é definido como valor REAL (flutuante) no software Connected Components Workbench. Para aprender mais sobre conversões e arredondamento de valores REAIS, consulte [Resolução de dados reais na página 177](#).

Velocidade inicial/parada do eixo

Velocidade inicial/parada é a velocidade inicial quando um eixo começa a se mover, e a última velocidade antes que o eixo pare de se mover. Em geral, a velocidade inicial/parada é configurada em algum valor baixo, de modo que seja menor que a maioria das velocidades utilizadas no bloco de funções de movimento.

- Quando a velocidade desejada é menor que a velocidade inicial/parada, mude o eixo imediatamente para a velocidade desejada.
- Quando a velocidade desejada NÃO é menor que a velocidade inicial/parada, mude o eixo imediatamente para a velocidade inicial/parada.

Resolução de dados reais

Certos elementos de dados e propriedades de eixo utilizam formato de dados REAIS (formato de ponto flutuante de precisão simples). Dados reais possuem resolução de sete dígitos e valores inseridos pelo usuário que forem maiores que os sete dígitos serão convertidos. Consulte os exemplos na [Tabela 21](#).

Tabela 21 - Exemplos de conversão de dados REAIS

Valor inserido pelo usuário	Convertido para
0,12345678	0,1234568
1234,1234567	1234,123
12345678	1,234568E+07 (formato exponencial)
0,000012345678	1,234568E-05 (formato exponencial)
2147418166	2,147418+E09
-0,12345678	-0,1234568

Se o número de dígitos for maior que sete (7) e o oitavo dígito for maior ou igual a 5, então o 7º dígito é arredondado para cima. Por exemplo:

21474185 arredondado para 2,147419E+07
 21474186 arredondado para 2,147419E+07

Se o oitavo dígito for <5, não é feito arredondamento e o sétimo dígito permanece igual. Por exemplo:

21474181 arredondado para 2,147418E+07

Tabela 22 - Exemplos para configuração de movimento

Parâmetro	Valor real inserido pelo usuário	Valor convertido pelo Connected Components Workbench	Valor de erro na dica ⁽¹⁾
Pulsos por revolução	8388608	8388608 (sem conversão)	Os pulsos por revolução devem estar no intervalo de 0,0001 a 8388607 unidades definidas pelo usuário.
Upper Soft Limit	10730175	1,073018E+7	limite programado superior (Upper Soft Limit) deve ser maior que o limite programado inferior (Lower Soft Limit). A faixa é de 0 (exclusivo) a 1,073217E+07 unidades definidas pelo usuário.
Lower Soft Limit	-10730175	-1,073018E+7	limite programado inferior (Lower Soft Limit) deve ser maior que o limite programado superior (Upper Soft Limit). O intervalo é de -1,073217E+07 a 0 (exclusivo) unidades definidas pelo usuário.

(1) Na página de configuração do eixo no Connected Components Workbench, um campo de entrada com uma borda vermelha indica que o valor que havia sido inserido é inválido. Uma mensagem de dica permite que você saiba a faixa esperada de valores para o parâmetro. A faixa de valores apresentada nas mensagens de dica também é apresentada no formato de dados REAL.

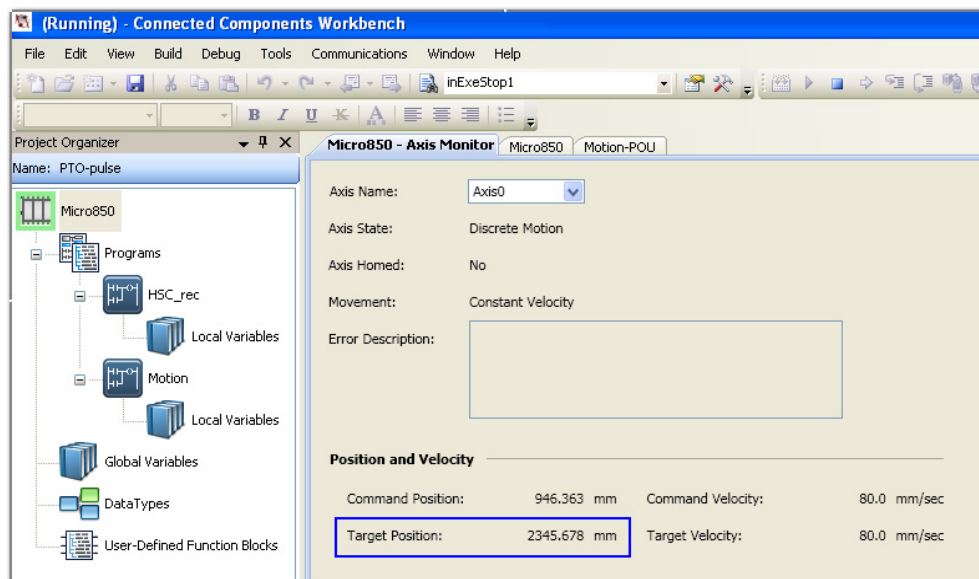
Exemplo de monitor de variáveis

O monitor de variáveis exibe seis dígitos significativos com arredondamento, porém o tipo de dados real ainda contém sete dígitos significativos.

Neste exemplo, o usuário inseriu o valor para Target Position (posição desejada) de 2345,678. Esse valor é arredondado para cima para seis dígitos (2345,68) na tela do monitoramento de variáveis.

Exemplo de monitor de eixo

O monitor de eixo exibe sete dígitos significativos com arredondamento.

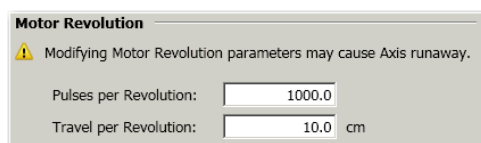


ATENÇÃO: Consulte [Configuração do eixo de movimento no Connected Components Workbench na página 169](#) para aprender mais sobre os diferentes parâmetros de configuração de eixo.

Precisão de pulso PTO

O recurso de movimento do Micro800 é baseado em pulso e o valor da distância e velocidade são projetados de modo que todos os valores relacionados a PTO são números inteiros no nível de hardware, ao converter para pulso PTO.

Por exemplo, se o usuário configura Motor Pulses per Revolution como 1.000 e Travel per Revolution como 10 cm e o usuário deseja gerar velocidade a 4,504 cm/s. A velocidade desejada é 4,504 cm/s (ou seja, 450,4 pulsos/s). Nesse caso, a velocidade real comandada será de 4,5 cm/s (ou seja, 450 pulsos/s), e o valor de 0,4 pulsos/s é eliminado por arredondamento.



Esse esquema de arredondamento também se aplica a outros parâmetros de entrada como Position, Distance, Acceleration, Deceleration, e Jerk. Por exemplo, com a configuração de revoluções de motor acima, configurar Jerk como 4,504 cm/s³ é o mesmo que configurar Jerk como 4,501 cm/s³, visto que ambos são arredondados para 4,5 cm/s³. Esse arredondamento aplica-se tanto à entrada de configuração do eixo no software Connected Components Workbench como à entrada do bloco de funções.

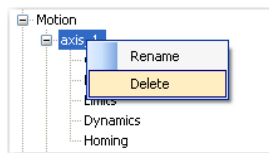
Validação de parâmetros de eixo de movimento

Além de permanecer no intervalo absoluto predeterminado, os parâmetros de eixo de movimento são validados com base nas relações com outros parâmetros. Essas relações ou regras estão listadas abaixo. O erro é marcado sempre que há violação dessas relações.

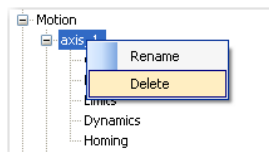
- limite programado inferior deve ser menor que o limite programado superior.
- Velocidade de arranque/parada não deve ser maior que a velocidade máxima.
- Velocidade de parada de emergência (Emergency Stop) não deve ser maior que a velocidade máxima.
- Velocidade de retorno à posição inicial não deve ser maior que a velocidade máxima.
- Homing Acceleration não deve ser maior que a aceleração máxima Maximum Acceleration.
- Homing Deceleration não deve ser maior que a desaceleração máxima Maximum Deceleration.
- Homing Jerk não deve ser maior que a variação de aceleração máxima Maximum Jerk.
- Homing Creep Velocity não deve ser maior que a velocidade máxima Maximum Velocity.

Excluir um eixo

1. Na árvore de configuração do dispositivo, e abaixo de Motion, clique com o botão direito no nome do eixo e selecione Delete.



2. Uma caixa de mensagem aparece pedindo para confirmar a exclusão. Clique em Yes.



Monitorar um eixo

Para monitorar um eixo, o software Connected Components Workbench deve ser conectado ao controlador e estar em modo de depuração (DEBUG).

1. Na página de configuração de dispositivo, clique em Axis Monitor.
2. A janela Axis Monitor aparece com as características a seguir disponíveis para visualização:
 - Estado do eixo
 - Eixo levado à posição inicial
 - Movimento
 - Descrição do erro

- Posição de comando em unidade definida pelo usuário
- Velocidade de comando em unidade definida pelo usuário por segundo
- Posição desejada em unidade definida pelo usuário
- Velocidade desejada em unidade definida pelo usuário por segundo

Bloco de funções de retorno à posição inicial

O bloco de funções de retorno à posição inicial MC_Home comanda o eixo a realizar a sequência de “buscar posição inicial”. A entrada “Position” é utilizada para configurar a posição absoluta quando o sinal de referência é detectado, e o offset inicial configurado é atingido. Este bloco de funções é concluído em “StandStill” se a sequência de movimento para posição inicial for bem-sucedida.

MC_Home pode ser abortado somente pelos blocos de funções MC_Stop ou MC_Power. Qualquer tentativa de abortar por meio de outros blocos de função de movimento resultará em falha do bloco de funções com ID de erro = MC_FB_ERR_STATE. Porém, a operação de retorno à posição inicial não é interrompida, e pode ser executada como normalmente ocorre.

Se MC_Home for abortado antes de ser concluído, a posição inicial buscada anteriormente é considerada como inválida, e o status Homed (levado à posição inicial) é removido.

Após a ligar o eixo, o status de levado à posição inicial (Homed) do eixo é resetado para o (não levado à posição inicial). Na maioria dos cenários, o bloco de funções MC_Home precisa ser executado para calibrar a posição do eixo de acordo com a posição inicial deste, configurada após a execução de MC_Power (On).

[Tabela 23 na página 181](#) descreve cinco modos de retorno à posição inicial suportados pelos controladores Micro830, Micro850 e Micro870.

Tabela 23 - Modos de retorno à posição inicial

Valor do modo de retorno à posição inicial	Nome do modo de retorno à posição inicial	Descrição do modo de retorno à posição inicial
0x00	MC_HOME_ABS_SWITCH	Processo de retorno à posição inicial busca pela chave de posição inicial absoluta (Home Absolute).
0x01	MC_HOME_LIMIT_SWITCH	Processo de retorno à posição inicial busca pela chave de fim de curso.
0x02	MC_HOME_REF_WITH_ABS	Processo de retorno à posição inicial busca pela chave de posição inicial absoluta (Home Absolute) além do uso de pulso de referência do encoder.
0x03	MC_HOME_REF_PULSE	Processo de retorno à posição inicial busca pela chave de fim de curso (Home Absolute) além do uso de pulso de referência do encoder.
0x04	MC_HOME_DIRECT	Processo de retorno estático à posição inicial com forçagem direta de uma posição inicial a partir de referência fornecida pelo usuário. O bloco de funções definirá a posição atual em que o mecanismo está como posição inicial, com sua posição determinada pelo parâmetro de entrada, “Position”.

IMPORTANTE Se o eixo estiver ativado com somente uma direção habilitada, o bloco de funções MC_Home (em modos 0, 1, 2, 3) irá gerar um erro e somente o bloco de funções MC_Home (modo 4) poderá ser executado. Consulte o bloco de funções MC_Power para mais detalhes.

Condições para retorno bem-sucedido à posição inicial

Para que a operação de retorno à posição inicial seja bem-sucedida, todas as chaves configuradas (ou sensores) precisam ser posicionados e cabeados apropriadamente. A ordem de posição correta a partir da posição mais negativa para a mais positiva – ou seja, da que fica mais para a esquerda até a que fica mais para a direita nos diagramas de configuração de retorno à posição inicial nesta seção – para as chaves é:

1. Chave de limite inferior
2. Chave de posição inicial ABS
3. Chave de limite superior

Durante a execução do bloco de funções MC_Home, a posição inicial será resetada, e a posição mecânica dos limites suaves será recalculada. Durante a sequência de retorno à posição inicial, a configuração de movimento configuração para os limites suaves será ignorada.

Assume-se os seguintes preceitos para as configurações da sequência de retorno à posição inicial discutida nesta seção:

1. Direção de retorno à posição inicial é configurada como direção negativa;
2. A chave de limite inferior é configurada como habilitada e cabeada;

Os diferentes modos de retorno à posição inicial conforme definido na [Tabela 23](#) podem ter sequências de movimento diferentes, mas ainda assim similares. O conceito discutido abaixo é aplicável a diversas configurações de retorno à posição inicial.

MC_HOME_ABS_SWITCH

IMPORTANTE Se a chave de posição inicial não está configurada como habilitada, retorno à posição inicial de MC_HOME_ABS_SWITCH (0) falha com MC_FB_ERR_PARAM.

Procedimento de retorno à posição inicial de MC_HOME_ABS_SWITCH (0) realiza uma operação de retorno à posição inicial contra a chave de posição inicial. A real sequência de movimento depende da configuração das chaves de posição inicial e fim de curso, depende também do status real para as chaves antes do retorno à posição inicial começar – ou seja, quando o bloco de funções MC_Home é executado.

Cenário 1: A peça móvel ao lado direito (positivo) da chave de posição inicial antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel se desloca para o lado esquerdo (direção negativa);
2. Quando a chave de posição inicial é detectada, a peça móvel desacelera até parar;
3. A peça móvel retrocede (direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de posição inicial;

4. Uma vez que o limiar ativada→desativada da chave de posição inicial tenha sido detectado, grave a posição como posição inicial mecânica, e desacelere até parar;
5. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento ao retroceder, mais o offset da posição inicial configurada para o eixo no software Connected Components Workbench.

Cenário 2: A peça móvel está entre as chaves de limite inferior e de posição inicial antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel se desloca para o seu lado esquerdo (direção negativa);
2. Quando a chave de limite inferior é detectada, a peça móvel desacelera até parar ou para imediatamente, segundo a configuração de parada física da chave de fim de curso;
3. A peça móvel retrocede (na direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de posição inicial;
4. Uma vez que o limiar ativada→desativada da chave de posição inicial tenha sido detectado, grave a posição como posição inicial mecânica, e desacelere até parar;
5. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento ao retroceder, mais o offset da posição inicial configurada para o eixo no software Connected Components Workbench.



Se a chave de limite inferior não está configurada ou cabeada, o movimento de retorno à posição inicial falha e ela move-se continuamente para a esquerda até que o inversor ou peça móvel falhe em mover-se.

Cenário 3: A peça móvel na chave de limite inferior ou chave de posição inicial antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel desloca-se para o seu lado direito (direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de posição inicial;
2. Uma vez que o limiar ativada→desativada da chave de posição inicial tenha sido detectado, grave a posição como posição inicial mecânica, e desacelere até parar;
3. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento para a direita, mais o offset da posição inicial configurada para o eixo no software Connected Components Workbench.

Cenário 4: A peça móvel ao lado esquerdo (negativo) da chave de limite inferior antes do retorno à posição inicial começar

Nesse caso, o movimento de retorno à posição inicial falha e move-se continuamente para a esquerda até que o inversor ou peça móvel falhe ao se mover. O usuário precisa certificar-se de que a peça móvel esteja no local apropriado antes do retorno à posição inicial começar.

MC_HOME_LIMIT_SWITCH

IMPORTANTE Se a chave de limite inferior não está configurada como habilitada, o retorno à posição inicial de MC_HOME_LIMIT_SWITCH (1) falhará (ID do erro: MC_FB_ERR_PARAM).

Para retorno à posição inicial contra a chave de limite inferior, um offset positivo de posição inicial pode ser configurado; para retorno à posição inicial contra a chave de limite superior, um offset negativo de posição inicial pode ser configurado.

Procedimento de retorno à posição inicial de MC_HOME_LIMIT_SWITCH (1) realiza uma operação de retorno à posição inicial contra a chave de fim de curso. A real sequência de movimento depende da configuração da chave de fim de curso e do status real para a chave antes do retorno à posição inicial começar – ou seja, quando o bloco de funções MC_Home é executado.

Cenário 1: A peça móvel ao lado direito (positivo) da chave de limite inferior antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel se desloca para o seu lado esquerdo (direção negativa);
2. Quando a chave de limite inferior é detectada, a peça móvel desacelera até parar ou para imediatamente, segundo a configuração de parada física da chave de fim de curso;
3. A peça móvel retrocede (direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de limite inferior;
4. Uma vez que o limiar ativada→desativada da chave de limite inferior tenha sido detectado, grave a posição como posição inicial mecânica, e desacelere até parar;
5. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento ao retroceder, mais o offset da posição inicial configurada para o eixo pelo software Connected Components Workbench.

Cenário 2: A peça móvel na chave fim de curso inferior antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel desloca-se para o seu lado direito (direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de limite inferior;
2. Uma vez que o limiar ativada→desativada da chave de limite inferior tenha sido detectado, grave a posição como posição inicial mecânica, e desacelere até parar;
3. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento para a direita, mais o offset da posição inicial configurada para o eixo pelo software.

Cenário 3: A peça móvel ao lado esquerdo (negativo) da chave de limite inferior antes do retorno à posição inicial começar

Nesse caso, o movimento de retorno à posição inicial falha e move-se continuamente para a esquerda até que o inversor ou peça móvel falhe ao se mover. O usuário precisa certificar-se de que a peça móvel esteja no local apropriado antes do retorno à posição inicial começar.

MC_HOME_REF_WITH_ABS

IMPORTANTE Se a chave de posição inicial ou pulso de ref. não estiver configurado como habilitado, retorno à posição inicial de MC_HOME_REF_WITH_ABS (2) falha com ID do erro: MC_FB_ERR_PARAM.

Procedimento de retorno à posição inicial MC_HOME_REF_WITH_ABS (2) realiza uma operação de retorno à posição inicial contra a chave de posição inicial, mais bom sinal de pulso de referência. A real sequência de movimento depende da configuração das chaves de posição inicial e fim de curso, depende também do status real para as chaves antes do retorno à posição inicial começar – ou seja, quando o bloco de funções MC_Home é executado.

Cenário 1: A peça móvel ao lado direito (positivo) da chave de posição inicial antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel se desloca para o seu lado esquerdo (direção negativa);
2. Quando a chave de posição inicial Abs é detectada, a peça móvel desacelera até parar;
3. A peça móvel retrocede (direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de posição inicial absoluta;
4. Uma vez que o limiar ativada→desativada da chave de posição inicial absoluta tenha sido detectado, comece a detectar o primeiro sinal de pulso de referência em entrada;
5. Quando o primeiro sinal de pulso de ref. vier, grave a posição como posição inicial mecânica e desacelere até parar;
6. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento ao retroceder, mais o offset da posição inicial configurada para o eixo pelo software Connected Components Workbench.

Cenário 2: A parte móvel entre a chave de limite inferior e chave de posição inicial antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel se desloca para o seu lado esquerdo (direção negativa);
2. Quando a chave de limite inferior é detectada, a peça móvel desacelera até parar ou para imediatamente, segundo a configuração de parada física da chave de fim de curso;

3. A peça móvel retrocede (na direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de posição inicial;
4. Uma vez que o limiar ativada→desativada da chave de posição inicial absoluta tenha sido detectado, comece a detectar o primeiro sinal de pulso de referência;
5. Quando o primeiro sinal de pulso de ref. vier, grave a posição como posição inicial mecânica e desacelere até parar;
6. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento ao retroceder, mais o offset da posição inicial configurada para o eixo pelo software Connected Components Workbench.

IMPORTANTE Nesse caso, se a chave de limite inferior não está configurada ou conectada, o movimento de retorno à posição inicial falhará e ela se moverá continuamente para a esquerda até que o inversor ou peça móvel falhe em mover-se.

Cenário 3: A peça móvel na chave de limite inferior ou chave de posição inicial antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel desloca-se para o seu lado direito (direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de posição inicial;
2. Uma vez que o limiar ativada→desativada da chave de posição inicial absoluta tenha sido detectado, comece a detectar o primeiro sinal de pulso de referência;
3. Quando o primeiro sinal de pulso de ref. vier, grave a posição como posição inicial mecânica e desacelere até parar;
4. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento para a direita, mais o offset da posição inicial configurada para o eixo no software Connected Components Workbench.

Cenário 4: A peça móvel ao lado esquerdo (negativo) da chave de limite inferior antes do retorno à posição inicial começar

Nesse caso, o movimento de retorno à posição inicial falha e move-se continuamente para a esquerda até que o inversor ou peça móvel falhe ao se mover. O usuário precisa certificar-se de que a peça móvel esteja no local apropriado antes do retorno à posição inicial começar.

MC_HOME_REF_PULSE

IMPORTANTE Se a chave de limite inferior ou pulso de ref. não estiver configurado como habilitado, retorno à posição inicial de MC_HOME_REF_PULSE (3) falha (ErrorID: MC_FB_ERR_PARAM).

Para retorno à posição inicial contra a chave de limite inferior, um offset positivo de posição inicial pode ser configurado; para retorno à posição inicial contra a chave de limite superior, um offset negativo de posição inicial pode ser configurado.

Procedimento de retorno à posição inicial MC_HOME_REF_PULSE (3) realiza uma operação de retorno à posição inicial contra a chave de fim de curso, mais bom sinal de pulso de referência. A real sequência de movimento depende da configuração da chave de fim de curso e do status real para as chaves antes do retorno à posição inicial começar – ou seja, quando o bloco de funções MC_Home é executado.

Cenário 1: A peça móvel ao lado direito (positivo) da chave de limite inferior antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel se desloca para o seu lado esquerdo (direção negativa);
2. Quando a chave de limite inferior é detectada, a peça móvel desacelera até parar ou para imediatamente, segundo a configuração de parada física da chave de fim de curso;
3. A peça móvel retrocede (direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de limite inferior;
4. Uma vez que o limiar ativada→desativada da chave de limite inferior tenha sido detectado, comece a detectar o primeiro sinal de pulso de referência;
5. Quando o primeiro sinal de pulso de ref. vier, grave a posição como a posição inicial mecânica e desacelere até parar;
6. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento ao retroceder, mais o offset da posição inicial configurada para o eixo pelo software Connected Components Workbench.

Cenário 2: A peça móvel na chave fim de curso inferior antes do retorno à posição inicial começar

A sequência de movimento de retorno à posição inicial para esse cenário é como descrito a seguir:

1. A peça móvel desloca-se para o seu lado direito (direção positiva) em velocidade de arraste para detectar o limiar ativada→desativada da chave de limite inferior;
2. Uma vez que o limiar ativada→desativada da chave de limite inferior tenha sido detectado, comece a detectar o primeiro sinal de pulso de referência;
3. Quando o primeiro sinal de pulso de ref. vier, grave a posição como a posição inicial mecânica e desacelere até parar;
4. Mova até a posição inicial configurada. A posição inicial mecânica registrada durante a sequência de movimento ao retroceder, mais o offset da posição inicial configurada para o eixo pelo software Connected Components Workbench.

Cenário 3: A peça móvel ao lado esquerdo (negativo) da chave de limite inferior antes do retorno à posição inicial começar

Nesse caso, o movimento de retorno à posição inicial falha e move-se continuamente para a esquerda até que o inversor ou peça móvel falhe ao se mover. O usuário precisa certificar-se de que a peça móvel esteja no local apropriado antes do retorno à posição inicial começar.

MC_HOME_DIRECT

O procedimento de retorno à posição inicial MC_HOME_DIRECT (4) realiza um retorno estático à posição inicial forçando diretamente uma posição real. Nenhum movimento físico é realizado neste modo. Isso é equivalente a uma ação MC_SetPosition, exceto pelo fato de que o status Axis Homed ficará ativado uma vez que MC_Home (modo = 4) seja executado com sucesso.

Use a PTO para controle PWM

O exemplo mostra como usar um eixo PTO como um PWM.

Inicie o Connected Components Workbench e crie o seguinte programa de lógica ladder.

Figura 12 - Exemplo 1: eixo PTO como um PWM

- 1 Habilite/energize o eixo do PWM imediatamente após ir para o modo de operação. O eixo PWM permanecerá energizado (até o modo de programação e assim por diante).

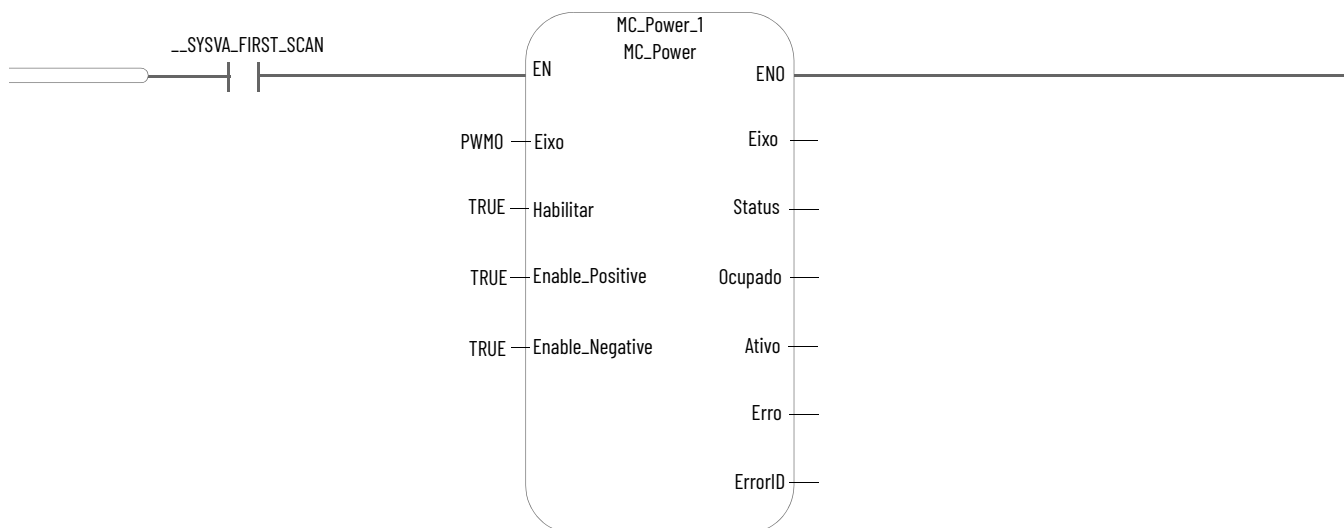


Figura 13 - Exemplo 2: eixo PTO como um PWM

- 2 Use continuamente MC_WriteParameter (parâmetro 1005) para mudar o ciclo de trabalho da variável global G_PWM_Duty_Cycle (exemplo: 0,5 => 50%)

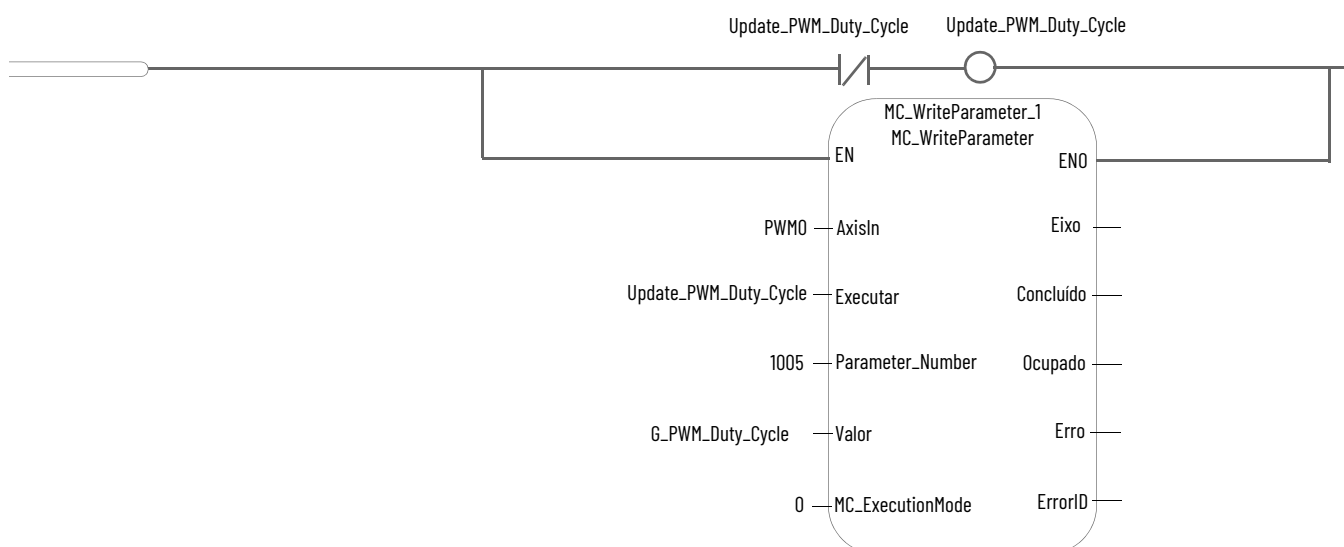
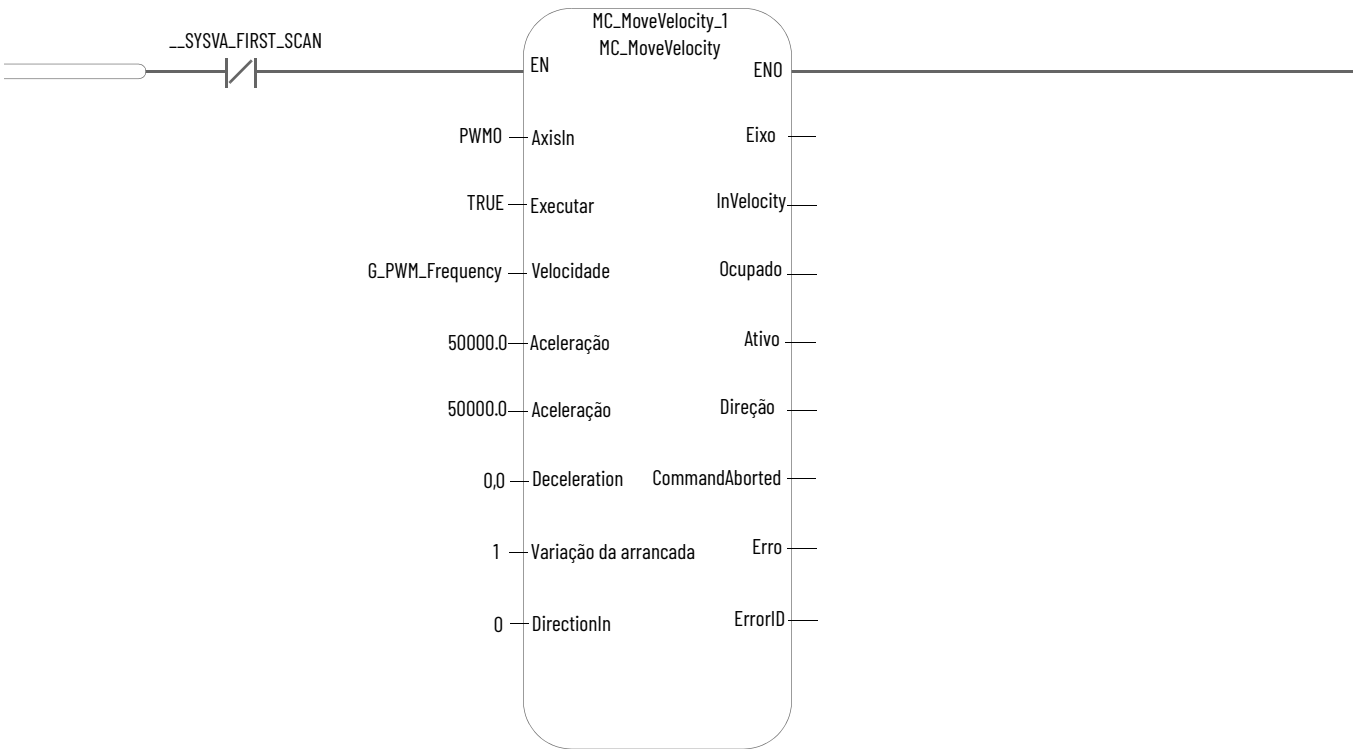


Figura 14 - Exemplo 3: eixo PT0 como um PWM

3 Após a primeira varredura, use MC_MoveVelocity para definir continuamente a frequência PWM (por exemplo: 50.000 => 50KHz) da variável global G_PWM_Frequency. O eixo PWM operará para sempre (até o modo de programa MC_Halt, e assim por diante).



POU PWM_Program

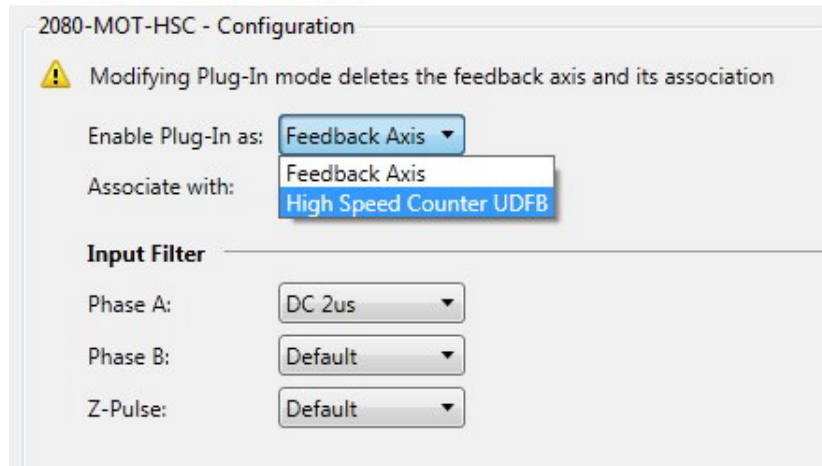
A POU define quatro variáveis.

Variável MC_Power_1 (* *) Direção: VAR Tipo de dados: MC_Power Atributo: ReadWrite Variável direta (canal):	Variable MC_MoveVelocity_1 (* *) Direção: VAR Tipo de dados: MC_MoveVelocity Atributo: ReadWrite Variável direta (canal):
Variable Update_PWM_Duty_Cycle (* *) Direção: VAR Tipo de dados: BOOL Atributo: ReadWrite Variável direta (canal):	Variável MC_Power_1 (* *) Direção: VAR Tipo de dados: MC_Power Atributo: ReadWrite Variável direta (canal):

Eixo de realimentação HSC

A partir do software Connected Components Workbench versão 8.0 e posteriores, foi adicionado suporte para um eixo de realimentação HSC (Contador de alta velocidade) que usa as mesmas instruções que o eixo de movimento PTO. UDFBs ainda são suportadas. Você pode usar qualquer um dos dois, mas não se pode selecionar ambos para o mesmo plug-in.

EXEMPLO: Exemplo de seleção do eixo de realimentação ou UDFB com plug-in 2080-MOT-HSC



O eixo de realimentação HSC oferece facilidade de uso, pois você já não precisa programar os blocos de função, e ele também usa menos memória no controlador. O eixo de realimentação HSC usa somente os blocos de função administrativos do eixo de movimento PTO e eles compartilham o mesmo monitor de eixo.

IMPORTANTE Os contadores não são zerado para o download do programa. Por exemplo, se usar o eixo de realimentação, use o bloco de funções MC_ResetPosition para zerar a posição.

IMPORTANTE Se o eixo de realimentação estiver em estado de erro porque os limites de posição configurados foram ultrapassados, o uso do bloco de funções MC_Reset para zer o eixo pode não remover o erro, pois pode ainda haver pulso detectado pelo encoder.

Use o contador de alta velocidade e a chave de fim de curso programável

Características gerais do contador de alta velocidade

Todos os controladores Micro830, Micro850 e Micro870, exceto o 2080-LCxx-AWB, suportam até seis contadores de alta velocidade (HSC). O recurso HSC no Micro800 consiste em dois componentes principais: o hardware do contador de alta velocidade (entradas incorporadas no controlador), e instruções do contador de alta velocidade no programa da aplicação. As instruções do contador de alta velocidade aplicam uma configuração ao hardware do contador de alta velocidade e atualizam o acumulador.



ATENÇÃO: Para utilizar o recurso HSC do Micro800 com eficiência, você precisa possuir um conhecimento básico do seguinte:

- Componentes HSC e elementos de dados.
As primeiras seções do capítulo fornecem uma descrição detalhada desses componentes. As instruções de início rápido (consulte a [página 265](#)) também estão disponíveis para orientá-lo pela configuração de um exemplo de projeto HSC.
- Programação e trabalho com elementos no software Connected Components Workbench.
O usuário precisa possuir um conhecimento de trabalho de programação por meio do diagrama de lógica ladder, texto estruturado, ou diagrama de blocos de funções para ser capaz de trabalhar com o bloco de funções e variáveis do HSC.



ATENÇÃO: Informações adicionais estão disponíveis sobre o bloco de funções HSC e seus elementos na ajuda online que acompanha a instalação do software Connected Components Workbench.

Este capítulo descreve como utilizar a função HSC e também contém seções sobre os blocos de funções HSC e HSC_SET_STS, como descrito a seguir:

- [Estruturas de dados do contador de alta velocidade \(HSC\)](#)
- [Bloco de funções HSC \(Contador de alta velocidade\)](#)
- [Bloco de funções HSC_SET_STS](#)
- [Função de chave de fim de curso programável \(PLS\)](#)
- [Interrupções HSC](#)

Características gerais da chave de fim de curso programável

A função chave de fim de curso programável permite configurar o contador de alta velocidade para operar como uma PLS (chave de fim de curso programável) ou uma chave de came rotativa. Para obter mais informações, consulte [Função de chave de fim de curso programável \(PLS\) na página 214](#).

O que é um contador de alta velocidade?

O contador de alta velocidade é utilizado para detectar e armazenar pulsos curtos (rápidos) e suas instruções especializadas para iniciar outras operações de controle baseadas em certas contagens atingirem valores pré-selecionados. Essas operações de controle incluem a execução automática e imediata da rotina de interrupção do contador de alta velocidade e a atualização imediata das saídas baseada em um padrão de fonte e máscara definido por você.

As funções do HSC são diferentes da maioria das outras instruções de controlador. Sua operação é realizada por circuitos padrão que funcionam em paralelo com o processador principal do sistema. Isso é necessário por causa das significativas exigências de desempenho dessas funções.

Recursos e operação

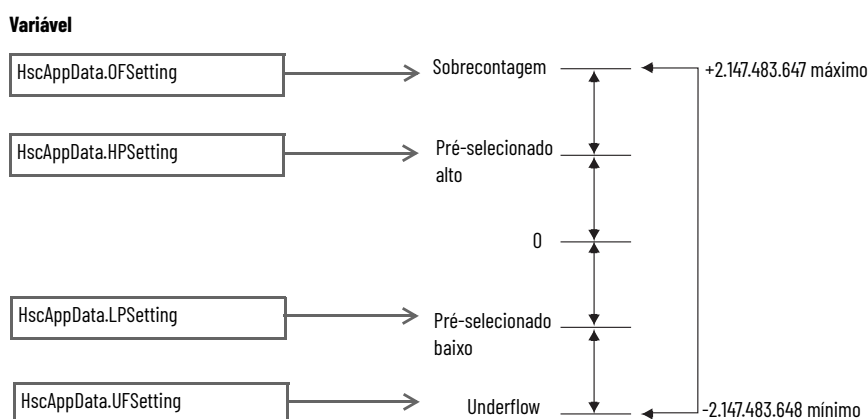
O HSC é extremamente versátil; você pode selecionar ou configurar o HSC mestre para qualquer um de dez (10) modos e o sub-HSC para qualquer um de cinco (5) modos de operação. Consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#) para obter mais informações.

Alguns dos recursos aprimorados dos contadores de alta velocidade são:

- Operação a 100 kHz
- Controle direto de saídas
- Dados inteiros com sinal e de 32 bits (faixa de contagem de $\pm 2.147.483.647$)
- Valores altos e baixos pré-selecionados programáveis, e referências de sobrecontagem e subcontagem
- Interrupção automática do processamento segundo a contagem acumulada
- Altere os parâmetros dinâmicos (do programa de controle do usuário)

A função de contador de alta velocidade opera como descrito no diagrama a seguir.

Figura 15 - Operação com contador de alta velocidade





Você precisa definir um valor adequado para as variáveis OFSetting, HPSetting, e UFSetting antes de disparar o início/execução HSC. Caso contrário, o controlador ficará com falha. (Definir um valor para LPSetting é opcional para determinados modos de contagem.)

Para aprender mais sobre a entrada variável HscAppData, consulte [Estrutura de dados HSC APP na página 196](#).

Ao utilizar blocos de funções HSC, recomenda-se que você:

- Defina a configuração de subcontagem HSCAppData (UFSetting) e configuração de valor baixo pré-selecionado (LPSetting) para um valor menor que o para evitar um possível mau funcionamento do HSC quando o acumulador deste for resetado para o.
- Defina a configuração de sobrecontagem HSCAppData (OFSetting) e configuração de valor alto pré-selecionado (HPSetting) para um valor maior que o para evitar um possível mau funcionamento do HSC quando o acumulador deste for resetado para o.

Em alguns casos, um subcontador será desabilitado pelo modo do contador mestre. Para obter mais informações, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).



HSC0 é utilizado neste documento para definir como qualquer HSC funciona.

IMPORTANTE A função HSC somente pode ser utilizada com as E/S incorporadas do controlador. Ela não pode ser usada com módulos de E/S de expansão.

Entradas HSC e mapeamento da fiação

Todos os controladores Micro830, Micro850 e Micro870, exceto 2080-LCxx-xxAWB, têm contadores de alta velocidade de 100 kHz. Cada contador de alta velocidade possui quatro entradas dedicadas e cada subcontador de alta velocidade possui duas entradas dedicadas.

Tabela 24 - Contadores de alta velocidade Micro830, Micro850 e Micro870

	10/16 pontos	24 pontos	48 pontos
Número de HSCs	2	4	6
Contadores de alta velocidade principais	1 (contador 0)	2 (contador 0,2)	3 (contadores 0, 2 e 4)
Subcontadores de alta velocidade	1 (contador 1)	2 (contador 1,3)	3 (contadores 1, 3 e 5)

Contador de alta velocidade	Entradas utilizadas
HSC0	0, 1, 2, 3
HSC1	2, 3
HSC2	4, 5, 6, 7
HSC3	6, 7
HSC4	8, 9, 10, 11
HSC5	10, 11

Subcontador da HSCo é HSC1, subcontador da HSC2 é HSC3 e subcontador da HSC4 é HSC5. Cada conjunto de contadores compartilha a mesma entrada. A tabela a seguir mostra as entradas dedicadas para os HSCs dependendo do modo.

Tabela 25 - Mapeamento da fiação das entradas HSC

	Entrada incorporada											
	0	01	02	03	04	05	06	07	08	09	10	11
HSC0	A/C	B/D	Rearme	Retenção								
HSC1			A/C	B/D								
HSC2					A/C	B/D	Rearme	Retenção				
HSC3							A/C	B/D				
HSC4									A/C	B/D	Rearme	Retenção
HSC5											A/C	B/D

As tabelas a seguir mostram o mapeamento da fiação de entrada para os diferentes controladores Micro830, Micro850 e Micro870.

Tabela 26 - Mapeamento da fiação de entrada HSC dos controladores Micro830 de 10 e 16 pontos

Modos de operação	Entrada 0 (HSC0) Entrada 2 (HSC1)	Entrada 1 (HSC0) Entrada 3 (HSC1)	Entrada 2 (HSC0)	Entrada 3 (HSC0)	Valor de modo no programa do usuário (HSCAppData.HSCMode)
Contador com direção interna (modo 1a)	Contagem crescente	Não usada			0
Contador com direção interna, reset externo e retenção (modo 1b)	Contagem crescente	Não usada	Rearme	Retenção	1
Contador com direção externa (modo 2a)	Contagem crescente/ decrecente	Direção	Não usada		2
Contador com direção externa, Reset e Retenção (modo 2b)	Contagem	Direção	Rearme	Retenção	3
Contador de duas entradas (modo 3a)	Contagem crescente	Contagem decrescente	Não usada		4
Contador de duas entradas com reset externo e retenção (modo 3b)	Contagem crescente	Contagem decrescente	Rearme	Retenção	5
Contador de quadratura (modo 4a)	Entrada tipo A	Entrada tipo B	Não usada		6
Contador de quadratura com reset externo e retenção (modo 4b)	Entrada tipo A	Entrada tipo B	Reset tipo Z	Retenção	7
Quadratura do contador X4 (modo 5a)	Entrada tipo A	Entrada tipo B	Não usada		8
Contador X4 de quadratura com reset externo e suporte	Entrada tipo A	Entrada tipo B	Reset tipo Z	Retenção	9

Tabela 27 - Mapeamento da fiação de entrada HSC dos controladores Micro830/Micro850/Micro870 de 24 pontos

Modos de operação	Entrada 0 (HSC0) Entrada 2 (HSC1) Entrada 4 (HSC2) Entrada 6 (HSC3)	Entrada 1 (HSC0) Entrada 3 (HSC1) Entrada 5 (HSC2) Entrada 7 (HSC3)	Entrada 2 (HSC0) Entrada 6 (HSC2)	Entrada 3 (HSC0) Entrada 7 (HSC2)	Valor de modo no programa do usuário
Contador com direção interna (modo 1a)	Contagem crescente	Não usada			0
Contador com direção interna, reset externo e retenção (modo 1b)	Contagem crescente	Não usada	Rearme	Retenção	1
Contador com direção externa (modo 2a)	Contagem crescente/ decrecente	Direção	Não usada		2

Tabela 27 - Mapeamento da fiação de entrada HSC dos controladores Micro830/Micro850/Micro870 de 24 pontos (Continuação)

Modos de operação	Entrada 0 (HSC0) Entrada 2 (HSC1) Entrada 4 (HSC2) Entrada 6 (HSC3)	Entrada 1 (HSC0) Entrada 3 (HSC1) Entrada 5 (HSC2) Entrada 7 (HSC3)	Entrada 2 (HSC0) Entrada 6 (HSC2)	Entrada 3 (HSC0) Entrada 7 (HSC2)	Valor de modo no programa do usuário
Contador com direção externa, Reset e Retenção (modo 2b)	Contagem crescente/decrescente	Direção	Rearme	Retenção	3
Contador de duas entradas (modo 3a)	Contagem crescente	Contagem decrescente	Não usada		4
Contador de duas entradas com reset externo e retenção (modo 3b)	Contagem crescente	Contagem decrescente	Rearme	Retenção	5
Contador de quadratura (modo 4a)	Entrada tipo A	Entrada tipo B	Não usada		6
Contador de quadratura com reset externo e retenção (modo 4b)	Entrada tipo A	Entrada tipo B	Reset tipo Z	Retenção	7
Quadratura do contador X4 (modo 5a)	Entrada tipo A	Entrada tipo B	Não usada		8
Contador X4 de quadratura com reset externo e suporte	Entrada tipo A	Entrada tipo B	Reset tipo Z	Retenção	9

Tabela 28 - Mapeamento da fiação de entrada HSC dos controladores Micro830/Micro850 de 48 pontos

Modos de operação	Entrada 0 (HSC0) Entrada 2 (HSC1) Entrada 4 (HSC2) Entrada 6 (HSC3) Entrada 8 (HSC4) Entrada 10 (HSC5)	Entrada 1 (HSC0) Entrada 3 (HSC1) Entrada 5 (HSC2) Entrada 7 (HSC3) Entrada 9 (HSC4) Entrada 11 (HSC5)	Entrada 2 (HSC0) Entrada 6 (HSC2) Entrada 10 (HSC4)	Entrada 3 (HSC0) Entrada 7 (HSC2) Entrada 11 (HSC4)	Valor de modo no programa do usuário
Contador com direção interna (modo 1a)	Contagem crescente	Não usada			0
Contador com direção interna, reset externo e retenção (modo 1b)	Contagem crescente	Não usada	Rearme	Retenção	1
Contador com direção externa (modo 2a)	Contagem crescente/decrescente	Direção	Não usada		2
Contador com direção externa, Reset e Retenção (modo 2b)	Contagem crescente/decrescente	Direção	Rearme	Retenção	3
Contador de duas entradas (modo 3a)	Contagem crescente	Contagem decrescente	Não usada		4
Contador de duas entradas com reset externo e retenção (modo 3b)	Contagem crescente	Contagem decrescente	Rearme	Retenção	5
Contador de quadratura (modo 4a)	Entrada tipo A	Entrada tipo B	Não usada		6
Contador de quadratura com reset externo e retenção (modo 4b)	Entrada tipo A	Entrada tipo B	Reset tipo Z	Retenção	7
Quadratura do contador X4 (modo 5a)	Entrada tipo A	Entrada tipo B	Não usada		8
Contador X4 de quadratura com reset externo e suporte	Entrada tipo A	Entrada tipo B	Reset tipo Z	Retenção	9

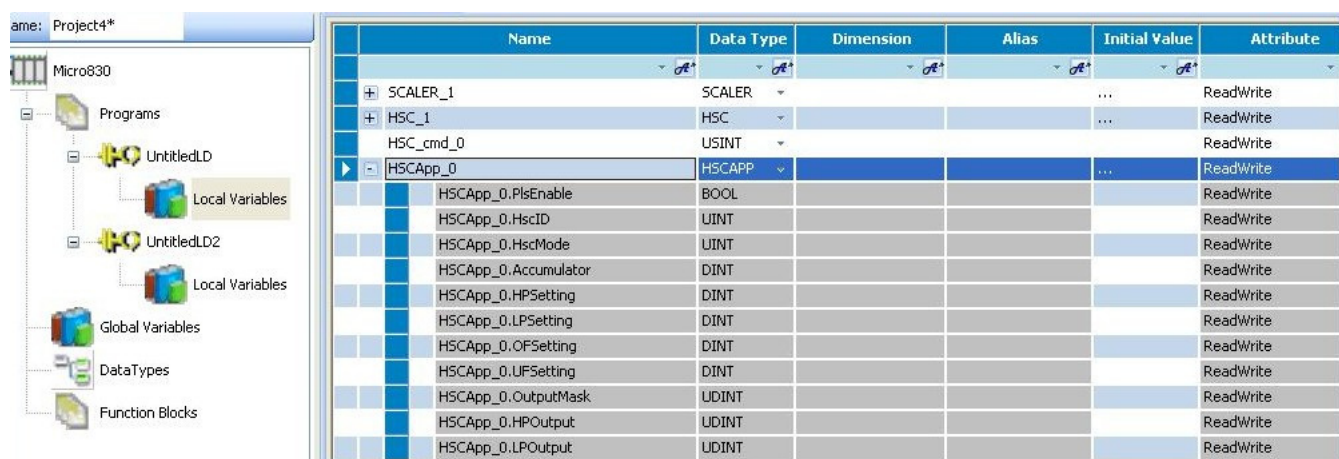
Estruturas de dados do contador de alta velocidade (HSC)

A seção a seguir descreve as estruturas de dados HSC.

Estrutura de dados HSC APP

Defina um dado HSC App (dados de configuração, tipo de dados HSCAPP) ao programar um HSC. Durante a contagem com HSC, você não deve alterar os dados, a não ser que precise recarregar a configuração.

Para recarregar a configuração HSC, mude os dados HSC APP, então chame o bloco de funções HSC com o comando OX03 (definir/recarregar). Caso contrário, a mudança para dados HSC App durante a contagem HSC é ignorada.



HSC1, HSC3, e HSC5 suportam os modos 0, 2, 4, 6 e 8 somente, enquanto HSC0, HSC2 e HSC4 suportam todos os modos de contagem.

Habilitação PLS (HSCAPP.PLSEnable)

Descrição	Formato dos dados	Acesso ao programa do usuário
PLSEnable	Bit	ler/gravar

Esse bit habilita e desabilita a função da chave de fim de curso programável HSC (PLS).

Quando a função PLS é habilitada, as configurações em

- HSCAPP.HpSetting
- HSCAPP.LpSetting
- HSCAPP.HpOutput
- HSCAPP.LpOutput

são suplantadas pelos valores de dados correspondentes dos dados PLS. Consulte [Função de chave de fim de curso programável \(PLS\) na página 214](#) para obter mais informações.

HSCID (HSCAPP.HSCID)

Descrição	Formato dos dados	Acesso ao programa do usuário
HscId	Palavra (UINT)	ler/gravar

[Tabela 29](#) lista as definições para HSCID.

Tabela 29 - Definição HSCID

Bits	Descrição
15 a 13	Tipo de módulo HSC: 0x00: Incorporado 0x01: Expansão (ainda não implementado) 0x02: Módulo plug-in
12 a 8	ID de slot de módulo: 0x00: Incorporado 0x01 a 0x1F: Expansão (ainda não implementado) 0x01 a 0x05: Módulo plug-in
7 a 0	ID HSC interna do módulo: 0x00-0x0F: Incorporado 0x00-0x07: Expansão (ainda não implementado) 0x00-0x07: Módulo plug-in

Para HSC incorporado, valor de HSCID válido é somente entre 0 a 5.

Modo HSC (HSCAPP.HSCMode)

Descrição	Formato dos dados	Acesso ao programa do usuário
Modo HSC	Palavra (UINT)	ler/gravar

A variável HSCMode define o contador de alta velocidade para um entre 10 tipos de operação. Esse valor inteiro é configurado por meio do dispositivo de programação e é acessível no programa de controle.

Modos de operação HSC

Número do modo	Type [Tipo]
0	Contador crescente – O acumulador é imediatamente desenergizado (0) quando alcança o pré-selecionado alto. Um pré-selecionado baixo não pode ser definido neste modo.
1	Contador crescente com reset e suporte externos – O acumulador é imediatamente desenergizado (0) quando alcança o pré-selecionado alto. Um pré-selecionado baixo não pode ser definido neste modo.
2	Contador com direção externa
3	Contador com direção externa, reset e retenção
4	Contador de duas entradas (para cima e para baixo)
5	Contador de duas entradas (para cima e para baixo) com reset externo e suporte
6	Contador de quadratura (entradas de fase A e B)
7	Contador de quadratura (entradas de fase A e B) com reset externo e suporte
8	Contador X4 de quadratura (entradas de fase A e B)
9	Contador X4 de quadratura (entradas de fase A e B) com reset externo e suporte

Os contadores de alta velocidade suportam 10 tipos de modo de operação e os subcontadores de alta velocidade suportam 5 tipos (modos 0, 2, 4, 6, 8). Se o contador principal de alta velocidade estiver configurado para modo 1, 3, 5, 7 ou 9, então o subcontador de alta velocidade será desabilitado.

Para mais informações sobre modos de operação da função HSC e atribuições de entrada, consulte [Entradas HSC e mapeamento da fiação na página 193](#).

Modo HSC 0 - Contador crescente

Tabela 30 - Exemplos de modo HSC 0

Terminais de entrada	Entrada incorporada 0				Entrada incorporada 1				Entrada incorporada 2				Entrada incorporada 3				Bit CE	Comentários
Função	Contagem				Não usada				Não usada				Não usada					
Exemplo 1	↑																ativado (1)	Acumulador HSC - 1 contagem
Exemplo 2	↑	ativado (1)	β	desativado (0)													desativado (0)	Manter o valor do acumulador

Células em branco = sem importância, ↑ = borda de subida, ↓ = borda de descida



As entradas de 0 a 11 estão disponíveis para utilização como entradas para outras funções independentemente do HSC sendo utilizado.

Modo HSC 1 - contador crescente com manter e reset externos

Tabela 31 - Exemplos de modo HSC 1

Terminais de entrada	Entrada incorporada 0				Entrada incorporada 1				Entrada incorporada 2				Entrada incorporada 3				Bit CE	Comentários
Função	Contagem				Não usada				Reset				Retenção					
Exemplo 1	↑								ativado (1)	↓	desativado (0)				desativado (0)		ativado (1)	Acumulador HSC - 1 contagem
Exemplo 2									ativado (1)	↓	desativado (0)		ativado (1)					Manter o valor do acumulador
Exemplo 3									ativado (1)	↓	desativado (0)						desativado (0)	Manter o valor do acumulador
Exemplo 4		ativado (1)	↓	desativado (0)					ativado (1)	↓	desativado (0)							Manter o valor do acumulador
Exemplo 5								↑										Limpar o acumulador (=0)

Células em branco = sem importância, ↑ = borda de subida, ↓ = borda de descida



As entradas de 0 a 11 estão disponíveis para utilização como entradas para outras funções independentemente do HSC sendo utilizado.

Modo HSC 2 - contador com direção externa

Tabela 32 - Exemplos de modo HSC 2

Terminais de entrada	Entrada incorporada 0				Entrada incorporada 1				Entrada incorporada 2				Entrada incorporada 3				Bit CE	Comentários
Função	Contagem				Direção				Não usada				Não usada					
Exemplo 1	↑							desativado (0)									ativado (1)	Acumulador HSC - 1 contagem
Exemplo 2	↑					ativado (1)											ativado (1)	Acumulador HSC - 1 contagem
Exemplo 3																	desativado (0)	Manter o valor do acumulador

Células em branco = sem importância, ↑ = borda de subida, ↓ = borda de descida



As entradas de 0 a 11 estão disponíveis para utilização como entradas para outras funções independentemente do HSC sendo utilizado.

Modo HSC 3 - contador com direção externa, reset e manter

Tabela 33 - Exemplos de modo HSC 3

Terminais de entrada	Entrada incorporada 0				Entrada incorporada 1				Entrada incorporada 2				Entrada incorporada 3				Bit CE	Comentários
Função	Contagem				Direção				Reset				Retenção					
Exemplo 1	↑							desativado (0)	ativado (1)	↓		desativado (0)				desativado (0)	ativado (1)	Acumulador HSC - 1 contagem
Exemplo 2	↑							ativado (1)	ativado (1)	↓		desativado (0)				desativado (0)	ativado (1)	Acumulador HSC - 1 contagem
Exemplo 3									ativado (1)	↓		desativado (0)				ativado (1)		Manter o valor do acumulador
Exemplo 4									ativado (1)	↓		desativado (0)					desativado (0)	Manter o valor do acumulador
Exemplo 5		ativado (1)	↓	desativado (0)					ativado (1)	↓		desativado (0)						Manter o valor do acumulador
Exemplo 6									↑									Limpar o acumulador (=0)

Células em branco = sem importância, ↑ = borda de subida, ↓ = borda de descida



As entradas de 0 a 11 estão disponíveis para utilização como entradas para outras funções independentemente do HSC sendo utilizado.

Modo HSC 4 - contador de duas entradas (crescente e decrescente)

Tabela 34 - Exemplos de modo HSC 4

Terminais de entrada	Entrada incorporada 0				Entrada incorporada 1				Entrada incorporada 2				Entrada incorporada 3				Bit CE	Comentários
Função	Contagem crescente				Contagem decrescente				Não usada				Não usada					
Exemplo 1	↑					ativado (1)	↓	desativado (0)									ativado (1)	Acumulador HSC - 1 contagem
Exemplo 2		ativado (1)	↓	desativado (0)	↑												ativado (1)	Acumulador HSC - 1 contagem
Exemplo 3																	desativado (0)	Manter o valor do acumulador

Células em branco = sem importância, ↑ = borda ascendente, ↓ = borda descendente



As entradas 0 a 11 estão disponíveis para utilização como entradas para outras funções independentemente do HSC sendo utilizado.

Modo HSC 5 - contador de duas entradas (crescente e decrescente) com manter e reset externos

Tabela 35 - Exemplos de modo HSC 5

Terminais de entrada	Entrada incorporada 0				Entrada incorporada 1				Entrada incorporada 2				Entrada incorporada 3				Bit CE	Comentários
Função	Contagem				Direção				Reset				Retenção					
Exemplo 1	↑					ativado (1)	↓	desativado (0)	ativado (1)	↓		desativado (0)				desativado (0)	ativado (1)	Acumulador HSC - 1 contagem
Exemplo 2		ativado (1)	↓	desativado (0)	↑				ativado (1)	↓		desativado (0)				desativado (0)	ativado (1)	Acumulador HSC - 1 contagem
Exemplo 3									ativado (1)	↓		desativado (0)				ativado (1)		Manter o valor do acumulador
Exemplo 4									ativado (1)	↓		desativado (0)					desativado (0)	Manter o valor do acumulador
Exemplo 5		ativado (1)	↓	desativado (0)					ativado (1)	↓		desativado (0)						Manter o valor do acumulador
Exemplo 6									↑									Limpar o acumulador (=0)

Células em branco = sem importância, ↑ = borda ascendente, ↓ = borda descendente



As entradas de 0 a 11 estão disponíveis para utilização como entradas para outras funções independentemente do HSC sendo utilizado.

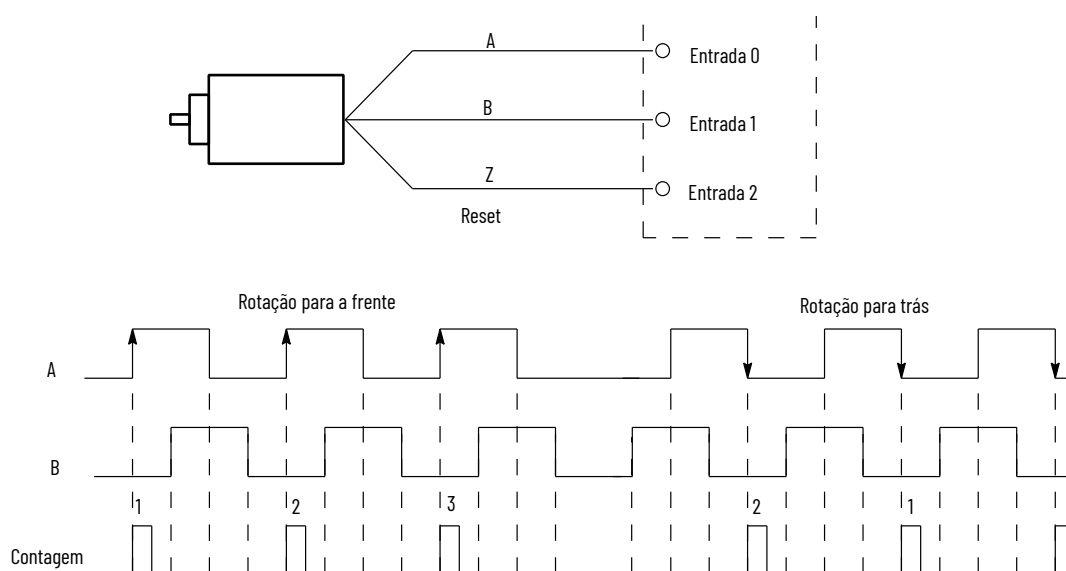
Usando o encoder de quadratura

O encoder de quadratura é utilizado para determinar a direção de rotação e posição para rotação em máquinas como um torno mecânico. O contador bidirecional conta a rotação do encoder de quadratura.

[Figura 16](#) mostra um encoder de quadratura conectado às entradas 0, 1 e 2. A direção de contagem é determinada pelo ângulo de fase entre A e B. Se A preceder B, o contador aumenta. Se B conduzir A, o contador diminui.

O contador pode ser atualizado utilizando a entrada Z. As saídas Z dos encoders tipicamente fornecem um pulso por revolução.

Figura 16 - Encoder de quadratura conectado às entradas



Modo HSC 6 - Contador de quadratura - entradas A e B faseadas

Tabela 36 - Exemplos de modo HSC 6

Terminais de entrada	Entrada incorporada 0				Entrada incorporada 1				Entrada incorporada 2				Entrada incorporada 3				Bit CE	Comentários
Função	Contagem A				Contagem B				Não usada				Não usada					
Exemplo 1 ⁽¹⁾	↑							desativado (0)									ativado (1)	Acumulador HSC - 1 contagem
Exemplo 2 ⁽²⁾			β					desativado (0)									ativado (1)	Acumulador HSC - 1 contagem
Exemplo 3				desativado (0)														Manter o valor do acumulador
Exemplo 4		ativado (1)																Manter o valor do acumulador
Exemplo 5						ativado (1)												Manter o valor do acumulador
Exemplo 6																	desativado (0)	Manter o valor do acumulador

(1) Entrada de contagem A precede a entrada de contagem B.

(2) Entrada de contagem B precede a entrada de contagem A.

Células em branco = sem importância, ↑ = borda ascendente, ↓ = borda descendente



As entradas de 0 a 11 estão disponíveis para utilização como entradas para outras funções independentemente do HSC sendo utilizado.

Modo HSC 7 - Contador de quadratura - entradas A e B faseadas com manter e reset externos

Tabela 37 - Exemplos de modo HSC 7

Terminais de entrada	Entrada incorporada 0				Entrada incorporada 1				Entrada incorporada 2				Entrada incorporada 3				Bit CE	Comentários
Função	Contagem A				Contagem B				Reset Z				Retenção					
Exemplo 1 ⁽¹⁾	↑						desativado (0)							desativado (0)			ativado (1)	Acumulador HSC - 1 contagem
Exemplo 2 ⁽²⁾			β				desativado (0)				desativado (0)			desativado (0)			ativado (1)	Acumulador HSC - 1 contagem
Exemplo 3			β	desativado (0)			desativado (0)		ativado (1)									Resete o acumulador a zero
Exemplo 4		ativado (1)																Manter o valor do acumulador
Exemplo 5						ativado (1)												Manter o valor do acumulador
Exemplo 6											desativado (0)		ativado (1)					Manter o valor do acumulador
Exemplo 7											desativado (0)						desativado (0)	Manter o valor do acumulador

(1) Entrada de contagem A precede a entrada de contagem B.

(2) Entrada de contagem B precede a entrada de contagem A. Células em branco = sem importância, ↑ = borda de subida, ↓ = borda descendente



As entradas de 0 a 11 estão disponíveis para utilização como entradas para outras funções independentemente do HSC sendo utilizado.

Modo HSC 8 - Contador de quadratura X4

Tabela 38 - Exemplos de modo HSC 8

Entrada incorporada 1 (HSC0) (A)	Entrada incorporada 1 (HSC0) (B)	Valor do bit CE	Acumulador e ação do contador
↑	OFF	TRUE	Ac. contagem crescente Valor
↑	ON	TRUE	Ac. contagem decrescente Valor
↓	OFF	TRUE	Ac. contagem decrescente Valor
↓	ON	TRUE	Ac. contagem crescente Valor
OFF	↑	TRUE	Ac. contagem decrescente Valor
ON	↑	TRUE	Ac. contagem crescente Valor
OFF	↓	TRUE	Ac. contagem crescente Valor
ON	↓	TRUE	Ac. contagem decrescente Valor
DESLIGADO ou LIGADO	DESLIGADO ou LIGADO	X	Ac. manter Valor
X	X	FALSE	Ac. manter Valor

Modo HSC 9 – Contador de quadratura X4 com manter e reset externos

Tabela 39 – Exemplos de modo HSC 9

Entrada incorporada 0 (HSC0) (A)	Entrada incorporada 1 (HSC0) (B)	Entrada incorporada 2 (HSC0) (reset)	Entrada incorporada 3 (HSC0) (manter)	Valor do bit CE	Acumulador e ação do contador
↑↑	OFF	X	-	TRUE	Ac. contagem crescente Valor
↑↑	ON	X	-	TRUE	Ac. contagem decrescente Valor
↓↓	OFF	X	-	TRUE	Ac. contagem decrescente Valor
↓↓	ON	X	-	TRUE	Ac. contagem crescente Valor
OFF	↑↑	X	-	TRUE	Ac. contagem decrescente Valor
ON	↑↑	X	-	TRUE	Ac. contagem crescente Valor
OFF	↓↓	X	-	TRUE	Ac. contagem crescente Valor
ON	↓↓	X	-	TRUE	Ac. contagem decrescente Valor
DESLIGADO ou LIGADO	DESLIGADO ou LIGADO	OFF	X	X	Ac. manter Valor
OFF	OFF	ON	X	X	Resete o acumulador a zero
X	X	OFF	ON	X	Ac. manter Valor
X	X	OFF	X	FALSE	Ac. manter Valor

Acumulador (HSCAPP.Accumulator)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCAPP.Accumulator	palavra longa (INT de 32 bits)	ler/gravar

Esse parâmetro é o valor do Accumulator HSC inicial que precisa ser definido ao iniciar o HSC. Esse parâmetro é atualizado pelo subsistema do HSC automaticamente quando o HSC está em modo de contagem, refletindo o valor real do acumulador do HSC.

Valor alto pré-selecionado (HSCAPP.HPSetting)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCAPP.HpSetting	palavra longa (INT de 32 bits)	ler/gravar

O HSCAPP.HPSetting é o ponto de ajuste superior (em contagens) que define quando o subsistema HSC gera uma interrupção.

Os dados carregados no valor alto pré-selecionado precisam ser menores que os dados residentes no parâmetro de sobrecontagem (HSCAPP.OFSetting), caso contrário um erro HSC é gerado.

Valor baixo pré-selecionado (HSCAPP.LPSetting)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCAPP.LpSetting	palavra longa (INT de 32 bits)	ler/gravar

O HSCAPP.LPSetting é o ponto de ajuste inferior (em contagens) que define quando o subsistema HSC gera uma interrupção.

Os dados carregados no valor baixo pré-selecionado devem ser:

1. Menor ou igual a 0 para os valores de parâmetro 0 e 1 do modo HSC (HSCAPP.HSCMode), caso contrário um erro HSC é gerado.

2. Maior ou igual aos dados residentes no parâmetro de subcontagem (HSCAPP.UFSetting) para todos os modos HSC (HSCAPP.HSCMode), caso contrário um erro HSC é gerado.

Se o valor de subcontagem e valores baixos pré-selecionados forem números negativos, o valor baixo pré-selecionado precisa ser um número com um valor absoluto menor.

Configuração de sobrecontagem (HSCAPP.OFSetting)

Descrição	Formato dos dados	Type [Tipo]	Acesso ao programa do usuário
HSCAPP.OFSetting	palavra longa (INT de 32 bits)	controle	ler/gravar

O HSCAPP.OFSetting define o limite de contagem superior para o contador. Se o valor acumulado do contador aumenta para além do valor especificado nessa variável, uma interrupção por sobrecontagem é gerada. Quando a interrupção por sobrecontagem é gerada, o subsistema HSC faz com que o acumulador mude para o valor de subcontagem e o contador continua contando a partir do valor de subcontagem (contagens não são perdidas nessa transição). O usuário pode especificar qualquer valor para a posição de sobrecontagem, desde que seja maior que o valor de subcontagem e fique entre -2.147.483.648 e 2.147.483.647.



Os dados carregados na variável de sobrecontagem precisam ser maiores que os dados residentes no valor alto pré-selecionado (HSCAPP.HPSSetting), caso contrário um erro HSC é gerado.

Configuração de subcontagem (HSCAPP.UFSetting)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCAPP.UFSetting	palavra longa (INT de 32 bits)	ler/gravar

O HSCAPP.UFSetting define o limite de contagem inferior para o contador. Se o valor acumulado do contador diminui abaixo do valor especificado nessa variável, uma interrupção por sobrecontagem é gerada. Quando a interrupção por subcontagem é gerada, o subsistema HSC reinicializa o valor acumulado para o valor de sobrecontagem e o contador então começa a contar a partir do valor de sobrecontagem (contagens não são perdidas nessa transição). O usuário pode especificar qualquer valor para a posição de subcontagem, desde que seja menor que o valor de sobrecontagem e fique entre -2.147.483.648 e 2.147.483.647.



Os dados carregados na variável de subcontagem precisam ser menores que ou iguais aos dados residentes no valor baixo pré-selecionado (HSCAPP.LPSSetting), caso contrário um erro HSC é gerado.

Bits de máscara de saída (HSCAPP.OutputMask)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCAPP.OutputMask	palavra (binária de 32 bits)	ler/gravar

O HSCAPP.OutputMask define quais saídas incorporadas no controlador podem ser diretamente controladas pelo contador de alta velocidade. O subsistema HSC possui a habilidade de ligar ou desligar diretamente (sem interação do programa de controle) as saídas no ato do acumulador HSC ter atingido os valores baixo ou alto pré-selecionados. O padrão de bits armazenado na variável HSCAPP.OutputMask define quais saídas são e quais não são controladas pelo HSC.

Por exemplo, se o usuário deseja controlar as saídas 0, 1 e 3 utilizando HSC, então o usuário precisa atribuir,
HscAppData.OutputMask = 2#1011
(OU usando valor decimal: HscAppData.OutputMask = 11)

O padrão de bits da variável HSCAPP.OutputMask corresponde diretamente aos bits de saída no controlador. Bits que são definidos como (1) são habilitados e podem ser ativados ou desativados pelo subsistema HSC. Bits que são limpos (0) não podem ser ativados ou desativados pelo subsistema HSC. O padrão de máscara de bits pode ser configurado somente durante a configuração inicial.

[Tabela 40](#) mostra um exemplo de como HPOutput e OutputMask controlam a saída incorporada.

Tabela 40 - Efeito da máscara de saída HSC nas saídas incorporadas

Variável de saída	Palavra de dados inteira com sinal e de 32 bits																				
	32 a 20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSCAPP.HPOutput (saída de valor alto pré-selecionado)		0	1	0	1	0	1	0	1	0	0	1	1	0	0	0	1	1	0	0	1
HSCAPP.OutputMask (máscara de saída)		1	1	0	0	0	0	0	0	0	1	1	0	0	0	1	1	0	0	1	1
Saída incorporada (de 10 pontos)																					
Saída incorporada (de 16 pontos)																					
Saída incorporada (de 24 pontos)																					
Saída incorporada (de 48 pontos)																					

As saídas mostradas nas caixas pretas são as saídas sob controle do subsistema HSC. A máscara define quais saídas podem ser controladas. Os valores das saídas de valor alto pré-selecionado ou valor baixo pré-selecionado (HSCAPP.HPOutput ou HSCAPP.LPOutput) definem se cada saída fica ATIVADA (1) ou DESATIVADA (0). Outro modo de observar isso é que a saída do valor baixo ou alto pré-selecionado é gravada por meio da máscara de saída, com a máscara de saída agindo como um filtro.

Os bits nas caixas cinza não estão sendo utilizados. Para o controlador de 10 pontos, os primeiros 4 bits da palavra de máscara são utilizados e os bits de máscara remanescentes não ficam funcionais, porque não se correlacionam a nenhuma saída física na unidade básica. Para os controladores de 16, 24 e 48 pontos, os primeiros 6, 10 e 20 bits da palavra de máscara são utilizados, respectivamente.

O padrão de máscara de bits pode ser configurado somente durante a configuração inicial.

Saída de valor alto pré-selecionado (HSCAPP.HPOutput)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCAPP.HPOutput	palavra longa (binário de 32 bits)	ler/gravar

A saída de valor alto pré-selecionado define o estado (1 = ATIVADA ou 0 = DESATIVADA) das saídas no controlador quando o valor alto pré-selecionado é atingido. Para mais informações sobre como ligar ou desligar diretamente as saídas ao atingir o valor alto pré-selecionado, consulte [Bits de máscara de saída \(HSCAPP.OutputMask\) na página 203](#).

O padrão de bits de saída de valor alto pré-selecionado pode ser configurado durante a configuração inicial, ou enquanto o controlador está em operação. Utilize o bloco de funções HSC para carregar os novos parâmetros enquanto o controlador está em operação.

Saída de valor baixo pré-selecionado (HSCAPP.LPOutput)

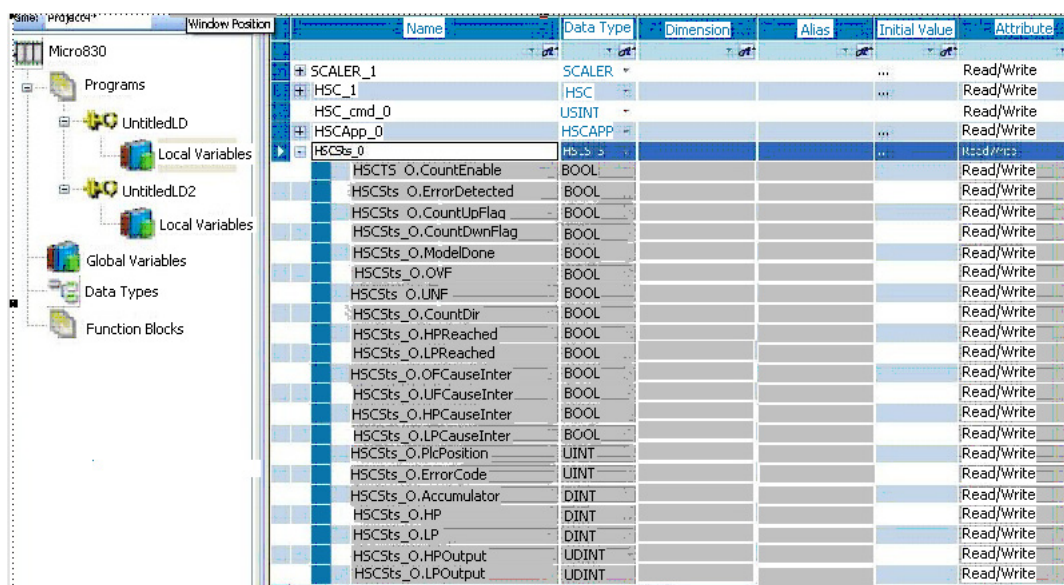
Descrição	Formato dos dados	Acesso ao programa do usuário
HSCAPP.LPOutput	palavra longa (binário de 32 bits)	ler/gravar

A saída de valor baixo pré-selecionado define o estado (1 = ATIVADA ou 0 = DESATIVADA) das saídas no controlador quando o valor baixo pré-selecionado é atingido. Consulte [Bits de máscara de saída \(HSCAPP.OutputMask\) na página 203](#) para mais informações sobre como energizar ou desenergizar diretamente as saídas ao atingir o valor baixo pré-selecionado.

O padrão de bits de saída de valor baixo pré-selecionado pode ser configurado durante a configuração inicial, ou enquanto o controlador está em operação. Utilize o bloco de funções HSC para carregar os novos parâmetros enquanto o controlador está em operação.

Estrutura de dados STS HSC (Status HSC)

Defina um dado STS HSC (dados de informação de status HSC, tipo de dado HSCSTS) ao programar um HSC.



Name	Data Type	Dimension	Alias	Initial Value	Attribute
SCALER_1	SCALER			...	Read/Write
HSC_1	HSC			...	Read/Write
HSC_cmd_0	USINT			...	Read/Write
HSCApp_0	HSCAPP			...	Read/Write
HSCSTS_0	HSCSTS			...	Read/Write
HSCSTS_0.CountEnable	BOOL			...	Read/Write
HSCSTS_0.ErrorDetected	BOOL			...	Read/Write
HSCSTS_0.CountUpFlag	BOOL			...	Read/Write
HSCSTS_0.CountDownFlag	BOOL			...	Read/Write
HSCSTS_0.ModelDone	BOOL			...	Read/Write
HSCSTS_0.OVF	BOOL			...	Read/Write
HSCSTS_0.UNF	BOOL			...	Read/Write
HSCSTS_0.CountDir	BOOL			...	Read/Write
HSCSTS_0.HPRReached	BOOL			...	Read/Write
HSCSTS_0.LPRReached	BOOL			...	Read/Write
HSCSTS_0.OFCauseInter	BOOL			...	Read/Write
HSCSTS_0.UFCauseInter	BOOL			...	Read/Write
HSCSTS_0.HPCauseInter	BOOL			...	Read/Write
HSCSTS_0.LPCauseInter	BOOL			...	Read/Write
HSCSTS_0.PlcPosition	UINT			...	Read/Write
HSCSTS_0.ErrorCode	UINT			...	Read/Write
HSCSTS_0.Accumulator	DINT			...	Read/Write
HSCSTS_0.HP	DINT			...	Read/Write
HSCSTS_0.LP	DINT			...	Read/Write
HSCSTS_0.HPOutput	UDINT			...	Read/Write
HSCSTS_0.LPOutput	UDINT			...	Read/Write

Contagem habilitada (HSCSTS.CountEnable)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.CountEnable	Bit	0 a 9	somente leitura

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O bit de controle de contagem habilitada é utilizado para indicar o status do contador de alta velocidade, em termos de contagem habilitada (1) ou desabilitada (0, padrão).

Erro detectado (HSCSTS.ErrorDetected)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.ErrorDetected	Bit	0 a 9	ler/gravar

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O flag de erro detectado é um bit de status que pode ser utilizado no programa de controle para detectar se um erro está presente no subsistema HSC. O tipo mais comum de erro que este bit representa é um erro de configuração. Quando esse bit é definido como (1), você deve observar o código de erro específico no parâmetro HSCSTS.ErrorCode. Esse bit é mantido pelo controlador e é definido quando há um erro HSC. Esse bit pode ser limpo pelo usuário, se necessário.

Contagem crescente (HSCSTS.CountUpFlag)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.CountUpFlag	Bit	0 a 9	somente leitura

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O bit de contagem crescente é utilizado com todos os HSCs (modos 0 a 9). Se o bit HSCSTS.CountEnable é definido, o bit de contagem crescente é definido como (1). Se o bit HSCSTS.CountEnable é limpo, o bit de contagem crescente é limpo (0).

Contagem decrescente (HSCSTS.CountDownFlag)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.CountDownFlag	Bit	2 a 9	somente leitura

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O bit de contagem decrescente é utilizado com os contadores bidirecionais (modos 2 a 9). Se o bit HSCSTS.CountEnable é definido, o bit de contagem decrescente é definido como (1). Se o bit HSCSTS.CountEnable é limpo, o bit de contagem decrescente é limpo (0).

Modo concluído (HSCSTS.Mode1Done)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.Mode1Done	Bit	0 ou 1	ler/gravar

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O flag de status concluído do modo é definido como (1) pelo subsistema HSC quando o HSC é configurado para comportamento de Modo 0 ou Modo 1, e o acumulador faz contagem crescente até o valor alto pré-selecionado.

Sobrecontagem (HSCSTS.OVF)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.OVF	Bit	0 a 9	ler/gravar

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O flag de status HSCSTS.OVF é definido como (1) pelo subsistema HSC sempre que o valor acumulado (HSCSTS.Accumulator) tiver contado pela variável de sobrecontagem (HSCAPP.OFSetting).

Esse bit é transicional e é definido pelo subsistema HSC. Depende do programa de controle utilizar, rastrear se necessário, e limpar (o) a condição de sobrecontagem.

Condições de sobrecontagem não geram uma falha no controlador.

Subcontagem (HSCSTS.UNF)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.UNF	Bit	0 a 9	ler/gravar

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O flag de status de subcontagem é definido como (1) pelo subsistema HSC sempre que a contagem do valor acumulado (HSCSTS.Accumulator) ficar menor que a variável de subcontagem (HSCAPP.UFSetting).

Esse bit é transicional e é definido pelo subsistema HSC. Depende do programa de controle utilizar, rastrear se necessário, e limpar (o) a condição de subcontagem.

Condições de subcontagem não geram uma falha no controlador.

Direção de contagem (HSCSTS.CountDir)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.CountDir	Bit	0 a 9	somente leitura

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O flag de status de direção de contagem é controlado pelo subsistema HSC. Quando o acumulador HSC faz contagem crescente, o flag de direção é definido como (1). Sempre que o acumulador HSC faz contagem decrescente, o flag de direção é limpo (o).

Se o valor acumulado para, o bit de direção mantém o seu valor. O único momento em que o flag de direção muda é quando a contagem acumulada é revertida.

Esse bit é atualizado continuamente pelo subsistema HSC sempre que o controlador está em um modo de operação.

Valor alto pré-selecionado atingido (HSCSTS.HPReached)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.HPReached	Bit	2 a 9	ler/gravar

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O flag de status de valor alto pré-selecionado atingido é definido como (1) pelo subsistema HSC sempre que o valor acumulado (HSCSTS.Accumulator) for maior que ou igual à variável de valor alto pré-selecionado (HSCAPP.HPSetting).

Esse bit é atualizado continuamente pelo subsistema HSC sempre que o controlador está em um modo de execução. Gravar nesse elemento não é recomendado.

Valor baixo pré-selecionado atingido (HSCSTS.LPReached)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.LPReached	Bit	2 a 9	somente leitura

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O flag de status de valor baixo pré-selecionado atingido é definido como (1) pelo subsistema HSC sempre que o valor acumulado (HSCSTS.Accumulator) for menor que ou igual à variável de valor baixo pré-selecionado (HSCAPP.LPSetting).

Esse bit é atualizado continuamente pelo subsistema HSC sempre que o controlador está em um modo de execução. Gravar nesse elemento não é recomendado.

Interrupção por sobrecontagem (HSCSTS.OFCauseInter)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.OFCauseInter	Bit	0 a 9	ler/gravar

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O bit de status de interrupção por sobrecontagem é definido como (1) quando o a contagem do acumulador HSC excede o valor de sobrecontagem e a interrupção HSC é disparada. Esse bit pode ser utilizado no programa de controle para identificar que a variável de sobrecontagem causou a interrupção HSC. Se o programa de controle precisa realizar qualquer ação de controle específica de acordo com a sobrecontagem, esse bit é utilizado como lógica condicional.

Esse bit pode ser limpo (0) pelo programa de controle e também é limpo pelo subsistema HSC sempre que essas condições são detectadas:

- É executada interrupção por valor baixo pré-selecionado
- É executada interrupção por valor alto pré-selecionado
- É executada interrupção por subcontagem

Interrupção por subcontagem (HSCSTS.UFCauseInter)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.UFCauseInter	Bit	2 a 9	ler/gravar

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O bit de status de interrupção por subcontagem é definido como (1) quando o a contagem do acumulador HSC excede o valor de subcontagem e a interrupção HSC é disparada. Esse bit pode ser utilizado no programa de controle para identificar que a condição de subcontagem causou a interrupção HSC. Se o programa de controle precisa realizar qualquer ação de controle específica de acordo com a subcontagem, esse bit é utilizado como lógica condicional.

Esse bit pode ser limpo (0) pelo programa de controle e também é limpo pelo subsistema HSC sempre que essas condições são detectadas:

- Ocorre interrupção por valor baixo pré-selecionado
- Ocorre interrupção por valor alto pré-selecionado
- Ocorre interrupção por sobrecontagem

Interrupção por valor alto pré-selecionado (HSCSTS.HPCauseInter)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.HPCauseInter	Bit	0 a 9	ler/gravar

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O bit de status de interrupção por valor alto pré-selecionado é definido como (1) quando o a contagem do acumulador HSC atinge o valor alto pré-selecionado e a interrupção HSC é disparada. Esse bit pode ser utilizado no programa de controle para identificar que a condição de valor alto pré-selecionado causou a interrupção HSC. Se o programa de controle precisa realizar qualquer ação de controle específica de acordo com o valor alto pré-selecionado, esse bit é utilizado como lógica condicional.

Esse bit pode ser limpo (0) pelo programa de controle e também é limpo pelo subsistema HSC sempre que essas condições são detectadas:

- Ocorre interrupção por valor baixo pré-selecionado
- Ocorre interrupção por subcontagem
- Ocorre interrupção por sobrecontagem

Interrupção segundo valor baixo pré-selecionado (HSCSTS.LPCauseInter)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.LPCauseInter	Bit	2 a 9	ler/gravar

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

O bit de status de interrupção por valor baixo pré-selecionado é definido como (1) quando o a contagem do acumulador HSC atinge o valor baixo pré-selecionado e a interrupção HSC tiver sido disparada. Esse bit pode ser utilizado no programa de controle para identificar que a condição de valor baixo pré-selecionado causou a interrupção HSC. Se o programa de controle precisa realizar qualquer ação de controle específica de acordo com o valor baixo pré-selecionado, esse bit seria utilizado como lógica condicional.

Esse bit pode ser limpo (0) pelo programa de controle e também é limpo pelo subsistema HSC sempre que essas condições são detectadas:

- Ocorre interrupção por valor alto pré-selecionado
- Ocorre interrupção por subcontagem
- Ocorre interrupção por sobrecontagem

Posição de chave de fim de curso programável (HSCSTS.PLSPosition)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.PLSPosition	palavra (INT)	0 a 9	somente leitura

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

Quando o HSC está em modo de contagem e PLS é habilitado, esse parâmetro indica qual elemento PLS é utilizado para a configuração HSC atual.

Código de erro (HSCSTS.ErrorCode)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCSTS.ErrorCode	palavra (INT)	0 a 9	somente leitura

(1) Para descrições de modo, consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#).

Os códigos de erro detectados pelo subsistema HSC são exibidos nessa palavra. Erros incluem:

Subelemento de código de erro	Código de erro de contagem HSC	Descrição do erro
Bit 15 a 8 (byte elevado)	0 a 255	O valor diferente de zero para o byte elevado indica que o erro HSC é devido à configuração de dados PLS. O valor do byte elevado indica qual elemento dos dados PLS dispara o erro.
Bit 7 a 0 (byte baixo)	0x00	Sem erro.
	0x01	Modo de contagem HSC inválido
	0x02	Valor alto pré-selecionado inválido
	0x03	Sobrecontagem inválida
	0x04	Subcontagem inválida
	0x05	Sem dados PLS

Gravar nesse elemento não é recomendado, exceto para remover erros existentes e captar novos erros HSC.

Acumulador (HSCSTS.Accumulator)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCSTS.Accumulator	palavra longa (INT de 32 bits)	somente leitura

HSCSTS.Accumulator contém o número de contagens detectadas pelo subsistema HSC. Se qualquer um dos modos 0 ou 1 for configurado, o acumulador é resetado para 0 quando um valor alto pré-selecionado é atingido ou quando uma condição de sobrecontagem é detectada.

Valor alto pré-selecionado (HSCSTS.HP)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCSTS.HP	palavra longa (INT de 32 bits)	somente leitura

O HSCSTS.HP é o ponto de ajuste superior (em contagens) que define quando o subsistema HSC gera uma interrupção.

Os dados carregados no valor alto pré-selecionado precisam ser menores que ou iguais aos dados residentes no parâmetro de sobrecontagem (HSCAPP.OFSetting), caso contrário um erro HSC é gerado.

Esse é a última configuração de valor alto pré-selecionado, que pode ser atualizada pela função PLS a partir do bloco de dados PLS.

Valor baixo pré-selecionado (HSCSTS.LP)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCSTS.LP	palavra longa (INT de 32 bits)	somente leitura

O HSCSTS.LP é o ponto de ajuste inferior (em contagens) que define quando o subsistema HSC gera uma interrupção.

Os dados carregados no valor baixo pré-selecionado precisam ser maiores que ou iguais aos dados residentes no parâmetro de subcontagem (HSCAPP.UFSetting), caso contrário um erro HSC é gerado. Se o valor de subcontagem e valores baixos pré-selecionados forem números negativos, o valor baixo pré-selecionado precisa ser um número com um valor absoluto menor.

Esse é a última configuração de valor baixo pré-selecionado, que pode ser atualizada pela função PLS a partir do bloco de dados PLS.

Saída de valor alto pré-selecionado (HSCSTS.HPOutput)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCSTS.HPOutput	palavra longa (binário de 32 bits)	somente leitura

A saída de valor alto pré-selecionado define o estado (1 = ATIVADA ou 0 = DESATIVADA) das saídas no controlador quando o valor alto pré-selecionado é atingido. Consulte [Bits de máscara de saída \(HSCAPP.OutputMask\) na página 203](#) para mais informações sobre como ligar ou desligar diretamente as saídas ao atingir o valor alto pré-selecionado.

Esse é a última configuração de saída de valor alto pré-selecionado, que pode ser atualizada pela função PLS a partir do bloco de dados PLS.

Saída de valor baixo pré-selecionado (HSCSTS.LPOutput)

Descrição	Formato dos dados	Acesso ao programa do usuário
HSCSTS.LPOutput	palavra longa (binário de 32 bits)	somente leitura

A saída de valor baixo pré-selecionado define o estado (1 = ATIVADA ou 0 = DESATIVADA) das saídas no controlador quando o valor baixo pré-selecionado é atingido. Consulte [Bits de máscara de saída \(HSCAPP.OutputMask\) na página 203](#) para mais informações sobre como energizar ou desenergizar diretamente as saídas ao atingir o valor baixo pré-selecionado.

Esse é a última configuração de saída de valor baixo pré-selecionado, que pode ser atualizada pela função PLS a partir do bloco de dados PLS.

Bloco de funções HSC (Contador de alta velocidade)

O bloco de funções HSC pode ser utilizado para iniciar/parar a contagem HSC, atualizar o status HSC, carregar novamente as configurações HSC, e resetar o acumulador HSC.

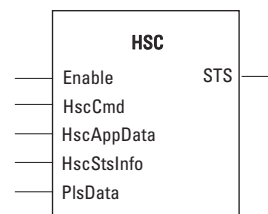


Tabela 41 - Parâmetros HSC

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição do parâmetro
Habilitar	Entrada	BOOL	Habilitar bloco de funções. Quando Enable = TRUE, realize a operação HSC especificada no parâmetro "HSC command". Quando Enable = FALSE, não há operação HSC, e não há atualização de status HSC.
HscCmd	Entrada	USINT	Consulte Comandos HSC na página 213
HscAppData	Entrada	Consulte Estrutura de dados HSC APP na página 196	Configuração da aplicação HSC. Geralmente, é necessária somente a configuração inicial.
PlsData	Entrada	Consulte vetor da Função de chave de fim de curso programável (PLS) na página 214	Dados da chave de fim de curso programável (PLS)
HscStsInfo	Saída	Consulte Estrutura de dados STS HSC (Status HSC) na página 205	Status dinâmico HSC. Informações de status geralmente são continuamente atualizadas durante a contagem HSC.
Sts	Saída	UINT	Status de execução de bloco de funções HSC

Comandos HSC (HscCmd)

HscCmd é um parâmetro de entrada com tipo de dado USINT. Todos os comandos HSC (1 a 4) são comandos de nível. Recomenda-se aos usuários desabilitar a instrução antes de atualizar o comando.

HscCmd = 1 inicia o mecanismo HSC. Uma vez que o HSC esteja em modo de operação, o **HscCmd = 2** precisa ser executado para que a contagem seja parada. Definir o parâmetro de entrada Enable como falso não para a contagem enquanto se estiver em modo de operação.

HscCmd = 3 recarrega os valores de parâmetro seguintes: HighPreset, LowPreset, Overflow, UnderFlow, HighPreset Output e LowPreset Output.

Os valores de parâmetro exibidos no monitor de variáveis podem não corresponder aos valores no hardware. Comando 3 precisa ser executado para carregar os valores das variáveis no hardware sem parar o HSC.

Se o HSC Enable for verdadeiro, HscCmd = 3 carregará os parâmetros continuamente. Dispare HscCmd = 3 somente uma vez.

HscCmd = 4 (reset) define o valor do acumulador para o valor de AppData.Accumulator do HSC. O HscCmd = 4 não para a contagem HSC. Se o HSC estiver contando quando o HscCmd = 4 for executado, algumas contagens podem ser perdidas.

Para resetar o valor do acumulador e então continuar a contagem, dispare o HscCmd = 4 somente uma vez. Se o comando é habilitado continuamente, ele pode causar erros.

O valor de AppData.Accumulator do HSC é atualizado automaticamente pelo mecanismo HSC com o mesmo valor do Sts.Accumulator do HSC. Para definir um valor específico para o acumulador HSC Acc durante a contagem, grave o valor para AppData.Accumulator do HSC imediatamente antes de executar HscCmd = 4.

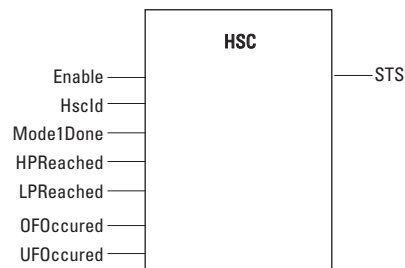
Tabela 42 - Comandos HSC

Comando HSC	Descrição
0x00	Reservado
0x01	OPERAÇÃO HSC <ul style="list-style-type: none"> • Inicie o HSC (se HSC estiver no modo inativo e a linha estiver habilitada) • Atualize somente as informações de status HSC (se o HSC já estiver no modo de operação e a linha estiver habilitada) • Atualize somente as informações de status HSC (se a linha estiver desabilitada)
0x02	Parada HSC: Pare uma contagem HSC (se o HSC estiver em modo de operação e a linha estiver habilitada)
0x03	Carga HSC: recarregue a configuração HSC (se a linha do programa estiver habilitada): HPSetting, LPSetting, HPOutput, LPOutput, OFSetting e UFSetting. acumulador HSC NÃO é recarregado pelo cmd = 0x03.
0x04	Reset de HSC: configure o acumulador para o valor atribuído, e resete as informações de status HSC (se a linha estiver habilitada)

Tabela 43 - Códigos de status de bloco de funções HSC

Código de status HSC	Descrição
0x00	Nenhuma ação a partir do controlador porque o bloco de funções não está habilitado
0x01	bloco de funções HSC executado com sucesso
0x02	Comando HSC inválido
0x03	ID HSC fora da faixa
0x04	Erro da configuração HSC

Bloco de funções HSC_SET_STS



O bloco de funções de status definido do HSC pode ser utilizado para alterar o status da contagem HSC. Esse bloco de funções é chamado quando o HSC não está em contagem (parado).

Tabela 44 - Parâmetros HSC

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição do parâmetro
Habilitar	Entrada	BOOL	Habilitar bloco de funções. Quando Enable = TRUE, defina/reseta o status HSC.. Quando Enable = FALSE, não existe mudança no status HSC.
HscId	Entrada	Consulte Estrutura de dados HSC APP na página 196	Descreve qual status HSC configurar.
Mode1Done	Entrada	BOOL	A contagem dos modos 1A ou 1B é realizada.

Tabela 44 - Parâmetros HSC (Continuação)

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição do parâmetro
HPReached	Entrada	BOOL	Valor alto pré-selecionado atingido Esse bit pode ser resetado para FALSE quando HSC não está em contagem.
LPReached	Entrada	BOOL	Valor baixo pré-selecionado atingido Esse bit pode ser resetado para FALSE quando HSC não está em contagem.
OF0ccurred	Entrada	BOOL	Ocorreu sobrecontagem. Esse bit pode ser resetado para FALSE quando necessário.
UF0ccurred	Entrada	BOOL	Ocorreu subcontagem. Esse bit pode ser resetado para FALSE quando necessário.
Sts	Saída	UINT	Status de execução de bloco de funções HSC Consulte Códigos de status de bloco de funções HSC na página 213 para descrição de código de status HSC (exceto 0x02 e 0x04).

Função de chave de fim de curso programável (PLS)

A função chave de fim de curso programável permite configurar o contador de alta velocidade para operar como uma PLS (chave de fim de curso programável) ou uma chave de came rotativa.

Quando a operação PLS está habilitada (HSCAPP.PLSEnable = Verdadeiro), o HSC (contador de alta velocidade) utiliza dados PLS para posições de limite/came. Cada posição de limite/came possui parâmetros de dados correspondentes que são utilizados para definir ou limpar saídas físicas na unidade básica do controlador. O bloco de dados PLS é ilustrado na [Figura 17 na página 215](#).

IMPORTANTE A função PLS opera somente em tandem com o HSC de um controlador Micro830. Para utilizar a função PLS, um HSC precisa primeiro ser configurado.

Estrutura de dados PLS

A função de chave de fim de curso programável é um conjunto adicional de modos de operação para o contador de alta velocidade (HSC). Ao operar nesses modos, os valores pré-selecionados e de saída são atualizados utilizando dados fornecidos pelo usuário cada vez em que um dos valores pré-selecionados é atingido. Esses modos são programados fornecendo um bloco de dados PLS que contenha os conjuntos de dados a serem utilizados.

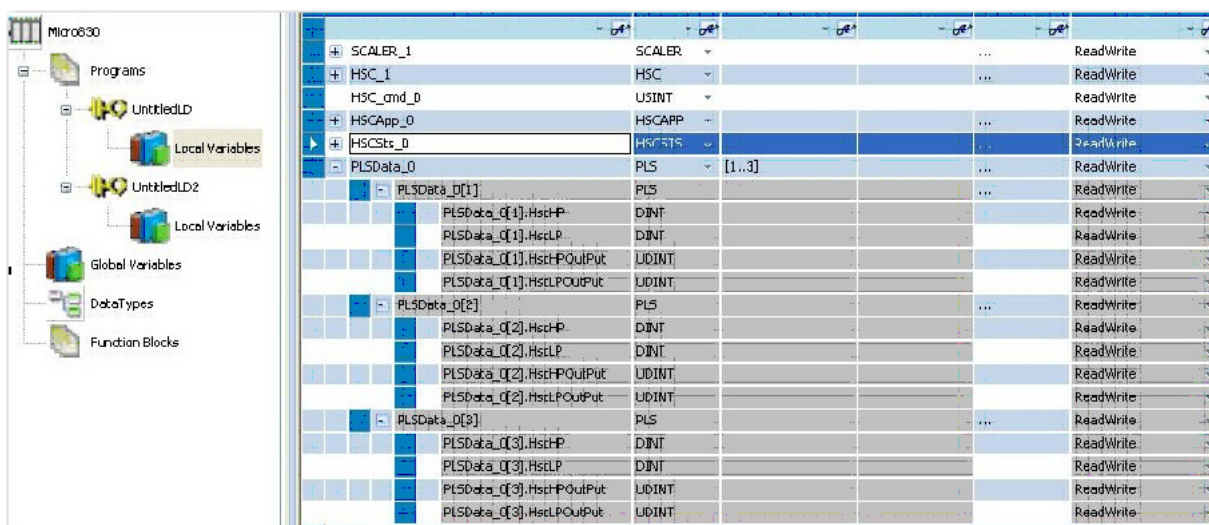
Estrutura de dados PLS é um vetor flexível, com cada elemento definido como demonstrado a seguir,

Ordem dos elementos	Tipo de dados	Descrição do elemento
Palavra 0 a 1	DINT	Definição de valor alto pré-selecionado
Palavra 2 a 3	DINT	Definição de valor baixo pré-selecionado
Palavra 4 a 5	UDINT	Dados de saída de valor alto pré-selecionado
Palavra 6 a 7	UDINT	Dados de saída de valor baixo pré-selecionado

O número total de elementos para um dado PLS não pode ser maior que 255.

Quando PLS não está habilitado, ainda é necessário definir dados PLS, mas não é necessário inicializá-los.

Figura 17 - Bloco de dados PLS



Operação PLS

Quando a função PLS está habilitada, e o controlador está em modo de operação, o HSC realiza contagem de pulsos em entrada. Quando a contagem atinge o primeiro valor pré-selecionado (HSCHP ou HSCLP) definido nos dados PLS, os dados de fonte de saída (HSCHPOutput ou HSCLPOutput) são gravados por meio da máscara HSC (HSCAPP.OutputMask).

Nesse ponto, os próximos valores pré-selecionados (HSCHP e HSCLP) definidos nos dados PLS tornam-se ativos.

Quando o HSC realiza contagem até esse novo valor pré-selecionado, os novos dados de saída são gravados por meio da máscara HSC. Esse processo continua até que o último elemento no bloco de dados PLS seja carregado. Nesse ponto, o elemento ativo no bloco de dados PLS é resetado para zero. Refere-se a esse comportamento como operação circular.



O HSCHPOutput é gravado somente quando HSCHP é atingido.
O HSCLPOutput é gravado quando HSCLP é atingido.



A saída High Data fica operacional somente quando o contador está em contagem crescente. A saída Low Data fica operacional somente quando o contador está em contagem decrescente.

Se dados inválidos forem carregados durante a operação, um erro HSC é gerado e causa uma falha no controlador.

Você pode utilizar o PLS em Up (elevado), Down (baixo) ou em ambas as direções. Se a sua aplicação contar somente em uma direção ignore os outros parâmetros.

A função PLS pode operar com todos os outros recursos HSC. A habilidade de selecionar quais eventos HSC geram uma interrupção de usuário sem limitações.

Exemplo PLS

Definindo os dados PLS

Utilizando o software Connected Components Workbench, defina a dimensão de HSC_PLS dos dados PLS como [1 a 4].

Definição dos dados PLS

Dados	Descrição	Formato dos dados
HSCHP	Pré-selecionado alto	Inteiro com sinal e de 32 bits
HSCLP	Pré-selecionado baixo	
HSCHPOutput	Saída High Data	Binário de 32 bits (bit 31-> 0000 0000 0000 0000 0000 0000 0000 <-bit 0)
HSCLPOutput	Saída Low Data	

Name	Data Type	Dimension	Initial Value	Attribute
HSC_1	HSC		...	ReadWrite
HSC_STS	HSCSTS		...	ReadWrite
HSC_APP	HSCAPP		...	ReadWrite
HSC_PLS	PLS	[1..4]	...	ReadWrite
HSC_PLS[1]	PLS		...	ReadWrite
HSC_PLS[1].HscHP	DINT		250	ReadWrite
HSC_PLS[1].HscLP	DINT		-2	ReadWrite
HSC_PLS[1].HscHPOutPut	UDINT		3	ReadWrite
HSC_PLS[1].HscLPOutPut	UDINT		0	ReadWrite
HSC_PLS[2]	PLS		...	ReadWrite
HSC_PLS[2].HscHP	DINT		500	ReadWrite
HSC_PLS[2].HscLP	DINT		-2	ReadWrite
HSC_PLS[2].HscHPOutPut	UDINT		7	ReadWrite
HSC_PLS[2].HscLPOutPut	UDINT		0	ReadWrite
HSC_PLS[3]	PLS		...	ReadWrite
HSC_PLS[3].HscHP	DINT		750	ReadWrite
HSC_PLS[3].HscLP	DINT		-2	ReadWrite
HSC_PLS[3].HscHPOutPut	UDINT		15	ReadWrite
HSC_PLS[3].HscLPOutPut	UDINT		0	ReadWrite
HSC_PLS[4]	PLS		...	ReadWrite
HSC_PLS[4].HscHP	DINT		1000	ReadWrite
HSC_PLS[4].HscLP	DINT		-2	ReadWrite
HSC_PLS[4].HscHPOutPut	UDINT		31	ReadWrite
HSC_PLS[4].HscLPOutPut	UDINT		0	ReadWrite

Uma vez que os valores acima para todos os 4 elementos de dados PLS tiverem sido inseridos, a PLS estará configurada.

Assuma que HSCAPP.OutputMask = 31 (o mecanismo HSC controla somente a saída incorporada o a 4), e HSCAPP.HSCMode = 0.

Operação PLS para este exemplo

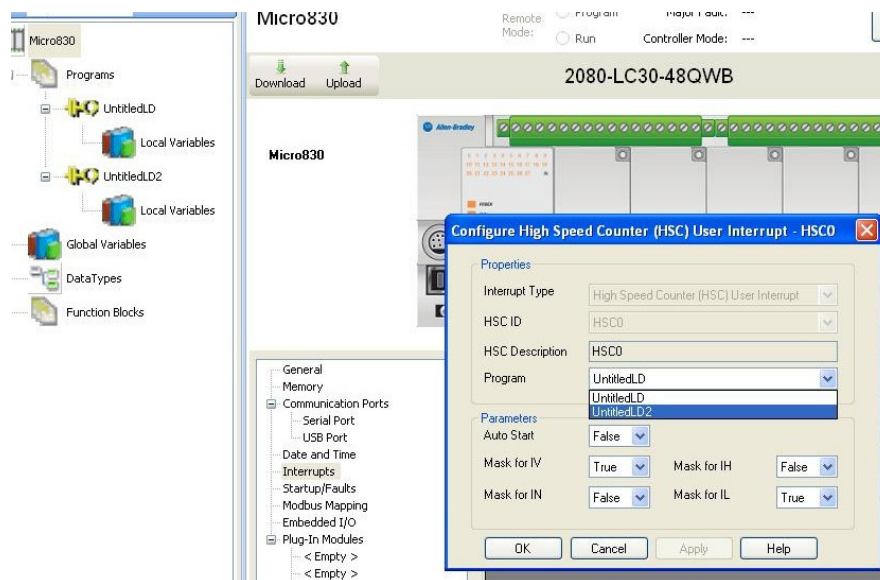
Quando a lógica ladder for executada pela primeira vez, HSCSTS.Accumulator = 1, portanto todas as saídas são desenergizadas. O valor de HSCSTS.HP = 250 Quando HSCSTS.Accumulator = 250, o HSC_PLS[1].HscHPOutput é enviado por meio do HSCAPP.OutputMask e energiza as saídas o e 1.

Isso se repetirá quando HSCSTS.Accumulator chegar a 500, 750 e 1000. O controlador alimenta as saídas o a 2, o a 3 e o a 4 respectivamente. Uma vez concluído, o ciclo é resetado e se repete a partir de HSCSTS.HP = 250.

Interrupções HSC

Uma interrupção é um evento que faz com que o controlador suspenda a tarefa que estiver realizando no momento, realize uma tarefa diferente, e então retorne à tarefa suspensa no ponto em que foi suspensa. Micro800 suporta até seis interrupções HSC.

Uma interrupção HSC é um mecanismo que os controladores Micro830, Micro850 e Micro870 fornecem para executar a lógica do usuário selecionada em um evento pré-configurado.



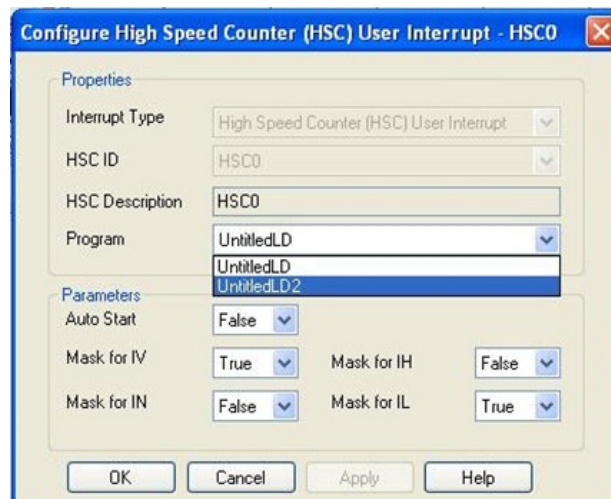
HSC0 é utilizado neste documento para definir como interrupções HSC funcionam.

Configuração de interrupção HSC

Na janela de configuração de interrupção de usuário, selecione HSC e então HSC ID, que é a interrupção que dispara a interrupção de usuário.

[Figura 18](#) mostra os campos selecionáveis na janela de configuração de interrupção.

Figura 18 - Janela de configuração de interrupção



POU da interrupção HSC

Esse é o nome da unidade organizacional de programa (POU) que é executada imediatamente quando essa interrupção HSC ocorre. Você pode escolher qualquer POU programada a partir da lista drop-down.

Início automático (HSCO.AS)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
AS – Início Automático	Bit	0 a 9	somente leitura

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O início automático é configurado com o dispositivo de programação e armazenado como parte do programa do usuário. O bit de início automático define se a função de interrupção HSC é iniciada automaticamente ou não sempre que o controlador entrar em qualquer modo de teste ou de operação.

Máscara para IV (HSCO.MV)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
MV – Máscara de sobrecontagem (Overflow Mask)	Bit	0 a 9	somente leitura

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O bit de controle de MV (máscara de sobrecontagem) é utilizado para habilitar (permitir) ou desabilitar (não permitir) a ocorrência de uma interrupção por sobrecontagem. Se esse bit estiver limpo (0) e uma condição de sobrecontagem atingida for detectada pelo HSC, a interrupção de usuário HSC não será executada.

Esse bit é controlado pelo programa do usuário e mantém o seu valor por meio de um ciclo de alimentação. É tarefa do programa do usuário definir e limpar esse bit.

Máscara para IN (HSCO.MN)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
MN – Máscara de subcontagem (Underflow Mask)	Bit	2 a 9	somente leitura

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O bit de controle de MN (máscara de subcontagem) é utilizado para habilitar (permitir) ou desabilitar (não permitir) a ocorrência de uma interrupção por subcontagem. Se esse bit estiver limpo (0) e uma condição de subcontagem atingida for detectada pelo HSC, a interrupção de usuário HSC não será executada.

Esse bit é controlado pelo programa do usuário e mantém o seu valor por meio de um ciclo de alimentação. É tarefa do programa do usuário definir e limpar esse bit.

Máscara para IH (HSCO.MH)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
MH – Máscara para valor alto pré-selecionado	Bit	0 a 9	somente leitura

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O bit de controle de MH (máscara de valor alto pré-selecionado) é utilizado para habilitar (permitir) ou desabilitar (não permitir) a ocorrência de uma interrupção por valor alto pré-selecionado. Se esse bit estiver limpo (0) e uma condição de valor alto pré-selecionado atingido for detectada pelo HSC, a interrupção de usuário HSC não será executada.

Esse bit é controlado pelo programa do usuário e mantém o seu valor por meio de um ciclo de alimentação. É tarefa do programa do usuário definir e limpar esse bit.

Máscara para IL (HSCO.ML)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
ML – Máscara para valor baixo pré-selecionado	Bit	2 a 9	somente leitura

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O bit de controle de ML (máscara de valor baixo pré-selecionado) é utilizado para habilitar (permitir) ou desabilitar (não permitir) a ocorrência de uma interrupção por valor baixo pré-selecionado. Se esse bit estiver limpo (0) e uma condição de valor baixo pré-selecionado atingido for detectada pelo HSC, a interrupção de usuário HSC não será executada.

Esse bit é controlado pelo programa do usuário e mantém o seu valor por meio de um ciclo de alimentação. É tarefa do programa do usuário definir e limpar esse bit.

Informação de status de interrupção HSC

Habilitação de interrupção de usuário (HSCO.Enabled)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCO.Enabled	Bit	0 a 9	somente leitura

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O bit habilitado é utilizado para indicar o status de habilitado ou desabilitado da interrupção HSC.

Execução de interrupção de usuário (HSCO.EX)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCO.EX	Bit	0 a 9	somente leitura

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O bit de EX (execução de interrupção de usuário) é definido como (1) sempre que o subsistema HSC iniciar o processamento da subrotina HSC devido a qualquer uma das seguintes condições:

- Valor baixo pré-selecionado atingido
- Valor alto pré-selecionado atingido
- Condição de sobrecontagem – contagem crescente por meio do valor de sobrecontagem
- Condição de subcontagem – contagem decrescente por meio do valor de subcontagem

O bit EX HSC pode ser utilizado no programa de controle como lógica condicional para detectar se uma interrupção HSC está em execução.

O subsistema HSC limpará (o) o bit EX quando o controlador completar o seu processamento da subrotina HSC.

Interrupção de usuário pendente (HSCO.PE)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCO.PE	Bit	0 a 9	somente leitura

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O PE (interrupção de usuário pendente) é um flag de status que representa que uma interrupção está pendente. Esse bit de status pode ser monitorado ou utilizado para propósitos de lógica no programa de controle se você precisar determinar quando uma subrotina não pode ser executada imediatamente. Esse bit é mantido pelo controlador e é definido e limpo automaticamente.

Interrupção de usuário perdida (HSCO.LS)

Descrição	Formato dos dados	Modos HSC ⁽¹⁾	Acesso ao programa do usuário
HSCO.LS	Bit	0 a 9	ler/gravar

(1) Para descrições do modo, consulte [Contagem decrescente \(HSCSTS.CountDownFlag\) na página 206](#).

O LS (interrupção de usuário perdida) é um flag de status que representa que uma interrupção foi perdida. O controlador pode processar 1 condição de interrupção do usuário ativa e manter até 1 condição de interrupção do usuário pendente antes de energizar o bit perdido.

Este bit é energizado pelo controlador. A utilização, rastreamento da condição de perda, se necessários, dependem do programa de controle.

Segurança do controlador

Segurança do Micro800 geralmente possui dois componentes:

- **Acesso exclusivo** que previne a configuração simultânea do controlador por dois usuários
- **Proteção com senha do controlador que protege a propriedade intelectual** contida no controlador e impede o acesso não autorizado

Modo protegido

Para manter a operação segura dos controladores Micro800, as operações que podem interromper a operação do módulo são restritas com base no modo de operação do módulo.

Tabela 45 - Operações protegidas nos controladores Micro800

Operação do controlador de corrente	Atividade						
	Solicitação de atualização do firmware	Ajuste de parâmetros da configuração da porta Ethernet ⁽¹⁾ (por meio do Connected Components Workbench ou RSLinx)	Alterações de configuração das portas serial e USB	Recuperação de senha perdida	Alteração de senha	Alteração do modo do controlador	Alteração de configuração de E/S
Controlador no modo de programa	Aceito		Não permitido	Aceito			Não permitido
Controlador com proteção de senha no modo de operação	Rejeitado	Não permitido ⁽²⁾	Não permitido	Rejeitado	Rejeitado	Rejeitado	Não permitido
Controlador em modo de operação forçada ⁽³⁾	Rejeitado	Não permitido	Não permitido	Rejeitado	Rejeitado	Rejeitado	Não permitido

(1) A configuração da Ethernet inclui endereço IP, máscara de sub-rede, gateway, velocidade/duplex da porta e assim por diante.

(2) A diferença entre Not Allowed e Rejected é que as atividades Not Allowed só podem ser feitas durante o off-line, enquanto as atividades Rejected podem ser executadas, mas não entram em vigor.

(3) O modo Hard Run pode estar apenas nos controladores Micro830, Micro850 e Micro870 com chave de modo para RUN.

Acesso exclusivo

Acesso exclusivo é aplicado ao controlador Micro800 independentemente dele ter ou não proteção por senha. Isso significa que somente uma sessão do software Connected Components Workbench é autorizada por vez e somente um cliente autorizado possui acesso exclusivo à aplicação do controlador. Isso garante que somente uma sessão de software possua acesso exclusivo à configuração específica da aplicação do Micro800.

O acesso exclusivo é aplicado nas versões 1 e 2 do firmware do Micro800. Quando um usuário do software Connected Components Workbench se conecta ao controlador Micro800, o controlador recebe acesso exclusivo àquele controlador.

Proteção por senha

Ao configurar uma senha no controlador, um usuário efetivamente restringe o acesso às conexões de software de programação ao controlador para sessões de software que forneçam a senha correta. Essencialmente, operações do software Connected Components Workbench como upload e download são impedidas se o controlador estiver protegido com uma senha e a senha correta não for fornecida.

Os controladores Micro800 com versão do firmware 2 e posterior são enviados sem senha definida. Mas, é possível definir uma senha por meio do software Connected Components Workbench (versão 2 ou posterior).

No software Connected Components Workbench versão 10 ou posterior, foi introduzido um algoritmo de senha mais forte para oferecer maior segurança. Para aproveitar todas as vantagens deste aprimoramento, o controlador Micro800 deve ter a versão do firmware 10 ou posterior e o projeto deve também ser da versão de software 10 ou superior.

No software Connected Components Workbench versão 20.01.00 ou posterior, o algoritmo de senha foi aprimorado para aumentar a criptografia de senha nos novos controladores Micro850 (2080-L50E) e Micro870 (2080-L70E). Quando a senha é alterada, ou removida e restaurada de volta no controlador, o código criptografado no controlador fica diferente. Portanto, a cópia de backup do programa no módulo 2080-MEMBAK-RTC2 deve ser atualizada antes que possa ser usada para restaurar o programa, caso contrário, a restauração irá falhar.

A senha do controlador também é gravada no módulo de backup de memória (2080-MEMBAK-RTC e 2080-MEMBAK-RTC2).

IMPORTANTE 2080-MEMBAK-RTC não é suportado pelos controladores Micro850 (2080-L50E) e Micro870.



Para instruções sobre como definir, alterar e remover senhas do controlador, consulte [Configure a senha do controlador na página 275](#).

Compatibilidade

O recurso de senha do controlador é suportado em:

- Software Connected Components Workbench, **versão 2** ou posterior
- Controladores Micro800 com firmware **de versão 2**

Para usuários com versões anteriores do software e/ou hardware (firmware), consulte os casos de compatibilidade abaixo.

Software Connected Components Workbench versão 1 com controlador Micro800
Versão do firmware 2

A conexão a um controlador Micro800 com versão de firmware 2 utilizando uma versão anterior do software Connected Components Workbench (versão 1) é possível e as conexões serão bem-sucedidas. Porém, o software não será capaz de determinar se o controlador está bloqueado ou não.

Se o controlador não está bloqueado o acesso à aplicação do usuário será permitido, desde que o controlador não esteja ocupado com outra sessão. Se o controlador estiver bloqueado, o acesso à aplicação do usuário irá falhar. Você precisa atualizar o software Connected Components Workbench para a versão 2.

Software Connected Components Workbench versão 2 com controlador Micro800 com versão de firmware 1

O software Connected Components Workbench versão 2 é capaz de “descobrir” e conectar-se a controladores Micro800 com versão de firmware anterior à versão 2 (ou seja, sem suporte ao recurso de senha de controlador). Porém, o recurso de senha de controlador não estará disponível para esses controladores. O usuário não será capaz de consultar interfaces associadas ao recurso de senha do controlador na sessão do software Connected Components Workbench.

Recomenda-se que os usuários atualizem o firmware. Consulte [Faça o upgrade flash do firmware de seu Micro800 na página 265](#) para obter instruções.



ATENÇÃO: O software Connected Components Workbench versão 9 ou anterior com controlador Micro800 com versão do firmware 10 ou posterior. Se um controlador Micro800 com versão do firmware 10 ou posterior for bloqueado usando o novo algoritmo de senha apresentado no software Connected Components Workbench versão 10 ou posterior, ele não poderá ser acessado usando o software Connected Components Workbench versão 9 ou anterior. Recomenda-se que os usuários atualizem para a mais recente versão do software Connected Components Workbench.

Trabalhar com um controlador bloqueado

Os fluxos de trabalho a seguir são suportados em controladores Micro800 compatíveis (versão de firmware 2) e software Connected Components Workbench versão 2

Carregar a partir de um controlador protegido com senha

1. Execute o software Connected Components Workbench.
2. No Organizador de projetos, expanda o catálogo clicando no sinal +.
3. Selecione o controlador desejado.
4. Selecione Upload.
5. Quando solicitado, forneça a senha do controlador.

IMPORTANTE Quando usar o software Connected Components Workbench versão 9 ou anterior:

- Você não pode carregar um projeto de versão 10 ou posterior do controlador.
- Você pode carregar um projeto de versão 9 ou anterior do controlador se este foi descarregado para o controlador usando o software Connected Components Workbench versão 10 ou posterior, mas não pode se conectar on-line.

Depurar um controlador protegido por senha

Para depurar um controlador bloqueado, você precisa conectar-se ao controlador por meio do software Connected Components Workbench e fornecer a senha antes de poder proceder para a depuração.

1. Execute o software Connected Components Workbench.
2. No Organizador de projetos, expanda o catálogo clicando no sinal +.
3. Selecione o código de catálogo do seu controlador.
4. Quando solicitado, forneça a senha do controlador.
5. Construa e salve o seu projeto.
6. Depure.

Download para um controlador protegido por senha

1. Execute o software Connected Components Workbench.
2. Clique em Connect.
3. Selecione o controlador desejado.
4. Quando solicitado, forneça a senha do controlador.
5. Construa e salve o projeto, se necessário.
6. Clique em Download.
7. Clique em Disconnect.

IMPORTANTE Se o controlador tiver um projeto bloqueado por senha da versão 10 ou posterior, não será possível acessar o controlador usando o software Connected Workbench versão 9 ou anterior. Se você usar o Software Connected Components Workbench versão 10 ou posterior para fazer o download de um projeto da versão 9 ou anterior, a senha no controlador será automaticamente convertida para o algoritmo antigo.

IMPORTANTE Se o controlador tem um projeto bloqueado com senha da versão 9 ou anterior e você usar o Software Connected Components Workbench versão 10 ou posterior para fazer o download de um projeto da versão 10 ou posterior, a senha no controlador será convertida automaticamente para o novo algoritmo.

IMPORTANTE Se a comunicação for perdida durante o download, repita o download e certifique-se de que o controlador é protegido com senha.

Transfira o programa do controlador e proteja com senha o controlador que receberá os dados

Nesse cenário, o operador precisa transferir a aplicação do usuário a partir do controlador 1 (bloqueado) para outro controlador Micro800 com o mesmo código de catálogo. A transferência da aplicação do usuário é realizada por meio do software Connected Components Workbench realizando a carga a partir do controlador 1, e então alterando o controlador de destino no projeto Micro800 e fazendo na sequência o download para o controlador 2. Finalmente, o controlador 2 será bloqueado.

1. No Organizador de projetos, clique no ícone Descobrir. Aparece a caixa de diálogo Browse connections.

2. Selecione o controlador 1 desejado.
3. Quando solicitado, insira a senha para o controlador 1.
4. Construa e salve o projeto.
5. Clique em Disconnect.
6. Desligue a alimentação para o controlador 1.
7. Troque o hardware do controlador 1 com o do controlador 2.
8. Ligue a alimentação para o controlador 2.
9. Clique em Connect.
10. Selecione o controlador 2 desejado.
11. Clique em Download.
12. Bloqueie o controlador 2. Consulte [Configure a senha do controlador na página 275](#).

Fazer backup e restaurar um controlador protegido por senha

Nesse fluxo de trabalho, será feita uma cópia de segurança da aplicação do usuário a partir de um controlador Micro800 que está bloqueado para um dispositivo de memória plug-in.

1. No Organizador de projetos, clique no ícone Descobrir. Aparece a caixa de diálogo Browse connections.
2. Selecione o controlador desejado.
3. Quando solicitado, forneça a senha do controlador.
4. Faça cópia de segurança de conteúdos do controlador a partir do módulo de memória.
O projeto no módulo de memória está agora bloqueado com senha.
5. Remova o módulo de memória do controlador 1 e insira no controlador 2.
6. Restaure o conteúdo do módulo de memória para o controlador 2.
Esta operação somente terá sucesso se:
 - o controlador não tiver senha – o projeto pode ser restaurado para o controlador definindo a opção “Load on power up” para o módulo de memória para Load Always.
 - a senha do controlador for igual à senha do projeto.

IMPORTANTE Mesmo se a senha for igual, a operação de restauração falhará se o controlador ou projeto no módulo de memória estiver protegido usando o algoritmo de senha antigo, e o outro estiver protegido usando o algoritmo de senha novo. É possível atualizar o controlador usando a opção Reset para remover a senha antes de restaurar o projeto para o controlador.

IMPORTANTE No software Connected Components Workbench versão 20.01.00 ou posterior, se a mesma senha for removida e restaurada nos novos controladores Micro850 (2080-L50E) e Micro870 (2080-L70E), a cópia de backup do programa no módulo plug-in 2080-MEMBAK-RTC2 deve ser atualizada antes, para que possa ser usada para restaurar o programa. Caso contrário, a restauração falhará quando o código de criptografia interno for alterado, já que a senha foi removida.

Configurar a senha do controlador

Para definir, alterar e remover senhas do controlador, consulte as instruções no início rápido [Configure a senha do controlador na página 275](#).

IMPORTANTE Após criar ou modificar a senha do controlador, você precisa desligar o controlador para que a senha seja salva.

Recuperação de uma senha perdida

Se o controlador estiver protegido com uma senha e a senha for perdida, então ficará impossível acessar o controlador utilizando o software Connected Components Workbench.

Para fazer a recuperação, o controlador deve ser definido no modo de programação utilizando a chave seletora para controladores Micro830, Micro850 ou Micro870, o 2080-LCD para controladores Micro810 ou o 2080-REMLCD para o Micro820. A seguir, o ControlFlash poderá ser usado para atualizar o firmware, o que também limpará a memória do controlador. No Software Connected Components Workbench versão 10 ou posterior, a opção Reset deve ser selecionada para que a memória do controlador seja limpa durante a atualização do firmware. Se a opção Upgrade ou Downgrade estiver selecionada, a senha será mantida.



ATENÇÃO: O projeto no controlador será perdido mas pode-se fazer download de um novo projeto.

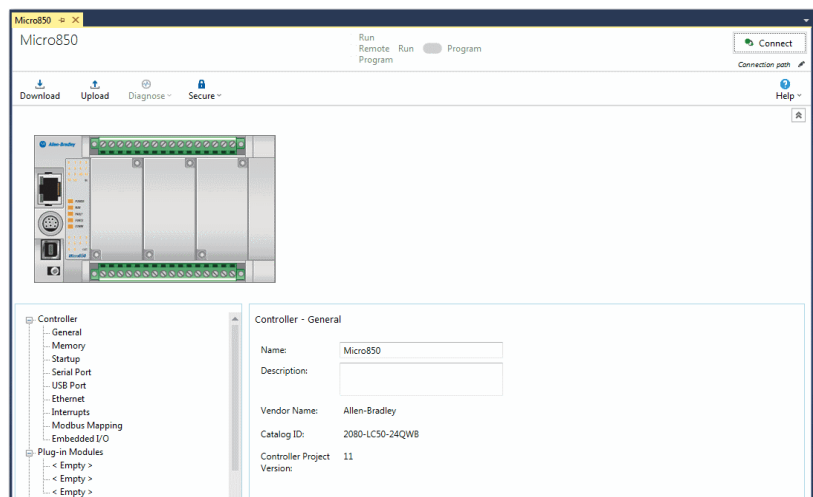
Usando o plug-in do módulo de memória

Você pode usar o módulo de memória (2080-MEMBAK-RTC e 2080-MEMBAK-RTC2) para baixar um programa em diferentes controladores.

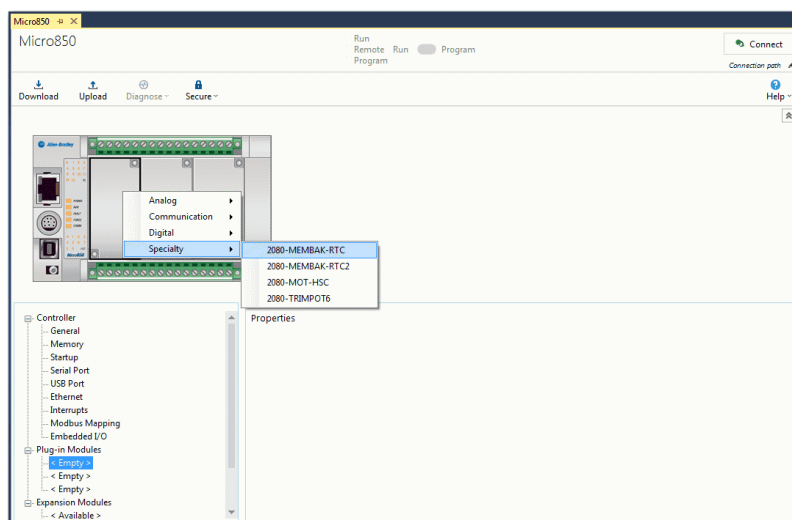
IMPORTANTE 2080-MEMBAK-RTC não é suportado pelos controladores Micro850 (2080-L50E) e Micro870.

Fazer backup do projeto

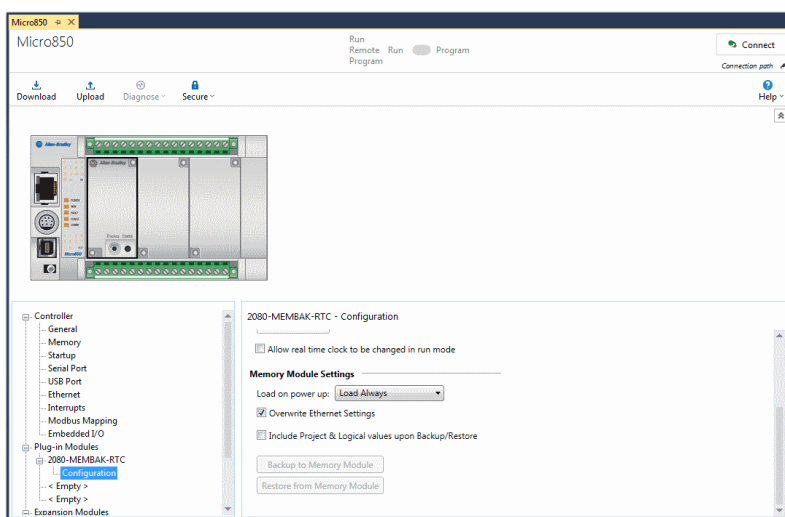
1. Para fazer um backup de projeto de um controlador para o módulo de memória, siga estas etapas:
Crie um novo programa e escolha o controlador desejado.
2. O controlador Micro850 é usado para este exemplo.



3. Clique duas vezes no ícone do controlador em Project Organizer para abrir a janela de propriedades do controlador.



4. Adicione o módulo de memória ao primeiro slot no controlador.



5. Clique em Configuration, nas propriedades MEMBAK-RTC e selecione “Load Always” ou “Load on Memory Error” para a opção Load on power up.
6. Construa e baixe o projeto para o controlador.

IMPORTANTE Enquanto conectado ao controlador, nas propriedades MEMBAK-RTC, certifique-se de que o controlador esteja alterado para o modo de programa e clique em “Backup to Memory Module” em Memory Module Settings. Selecione Sim para baixar o programa no Memory Module. Uma janela deve aparecer informando que a operação foi concluída com sucesso.

O status de Incompatibilidade de Senha deve estar em “False”, isso significa que o projeto de controlador e backup tem a mesma condição de segurança.

Se o status for “True”, o Controlador não carregará do módulo de Memória, pois a condição de segurança é incompatível.

Recuperar o projeto

1. Para restaurar o projeto do módulo de memória para o controlador, siga estas etapas:

Enquanto conectado ao controlador em propriedades MEMBAK-RTC, certifique-se de que o controlador esteja alterado para o modo de programa e clique em "Restore from Memory Module" em Memory Module Settings. Selecione Sim para baixar o programa no controlador. Uma janela deve aparecer informando que a operação foi concluída com sucesso.

Usando o módulo de memória para copiar um projeto para vários controladores

1. Você pode usar o módulo de memória para baixar um projeto para vários controladores sem conectá-los a um PC com o software Connected Components Workbench instalado. Para fazer isso:
2. Faça backup de um projeto com a opção "Load Always" ativada.
3. Remova o módulo e conecte-o a um controlador diferente.
4. Ligue a alimentação. Observe que o LED de status no módulo acende por alguns segundos enquanto o projeto está sendo baixado do módulo para o controlador.

Usando cartões microSD

Esse capítulo fornece uma descrição do suporte do cartão microSD para controladores Micro830, Micro850 e Micro870.

Tópico	Página
Características gerais	229
Fazer backup e restaurar projeto	230
Backup e restauração da estrutura do diretório	231
Ajustes de parâmetros de energização no ConfigMeFirst.txt	232
Regras gerais de configuração no ConfigMeFirst.txt	234
Erros do ConfigMeFirst.txt	234
Registro de dados	237
Receita	243
Projetos de início rápido para blocos de funções de registro de dados e de receita	247

A última seção fornece projetos de início rápido para as funções de registro de dados e receita.

Características gerais

Com a versão do firmware 12.011 e posteriores, os controladores Micro830, Micro850 e Micro870 suportam o uso de cartões microSD por meio do plug-in de cartão microSD (um produto de parceiro da PartnerNetwork™ Technology), para controladores Micro800 para as finalidades a seguir:

- Backup e restauração de projeto
- Registro de dados e receita

Recomendamos a utilização do cartão microSD 2080-SD-2GB da Allen-Bradley.

IMPORTANTE Para controladores Micro850 (2080-L50E) E Micro870 (2080-L70E), não use nenhum cartão microSD com menos de 2 GB de memória.

IMPORTANTE Para um desempenho ideal, o cartão microSD não deve estar mais de 90% cheio. Verifique regularmente o espaço disponível em seu cartão microSD e certifique-se de que o cartão seja usado exclusivamente para o controlador Micro800 e que nenhum arquivo desnecessário esteja presente. Exclua regularmente diretórios e arquivos de registro de dados antigos.

IMPORTANTE Não remova o cartão microSD ou desligue durante operações em andamento como carregar, descarregar, excluir, pesquisar, fazer backup e restaurar para evitar perda de dados. Um LED de status SD piscando indica que estas operações estão em andamento.

Observe o seguinte:

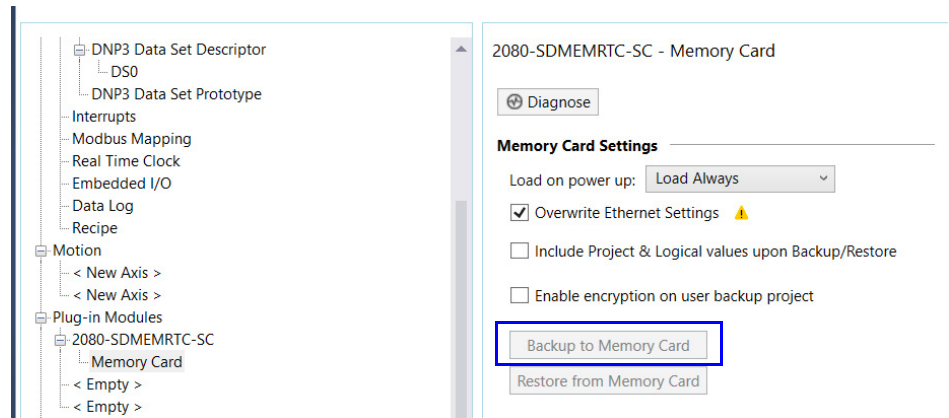
- O LED de status SD não piscará ao atualizar o firmware do cartão microSD.
- O LED de status SD não pisca continuamente durante toda a duração da operação de restauração.

IMPORTANTE Para evitar que os dados sejam perdidos, os blocos de funções de receita e registro de dados devem indicar o status Idle antes que o cartão microSD seja removido.

Fazer backup e restaurar projeto

O backup e a restauração do projeto nos controladores Micro830, Micro850 e Micro870 são suportados principalmente pelo cartão microSD. Tanto o backup quanto a restauração podem ser iniciados ou disparados manualmente e configurados pelo software Connected Components Workbench e pelo arquivo ConfigMeFirst.txt no cartão microSD. Esses arquivos de backup não são os mesmos que os arquivos de projeto do Connected Components Workbench.

O backup e a restauração podem ocorrer somente quando o controlador está no modo de PROGRAMA. Na energização do controlador, a restauração ocorrerá automaticamente se a opção Load Always ou Load on Memory Error tiver sido configurada no software Connected Components Workbench.



No software Connected Components Workbench versão 20.01.00 ou posterior, para os controladores Micro850 (2080-L50E) e Micro870 (2080-L70E) com o módulo plug-in de cartão microSD, foi adicionada a opção de funcionalidade de criptografia para ajudar a melhorar a capacidade de criptografia em seus programas para uma melhor proteção de segurança.

Se você selecionar a opção “Enable Encryption on user backup project”, o tempo que leva para restaurar o programa pode aumentar até 10 vezes, dependendo do tamanho e do conteúdo do programa.

IMPORTANTE Para saber mais sobre backup e restauração usando o software Connected Components Workbench, consulte a ajuda on-line do software.

IMPORTANTE Para controladores Micro800 que suportam cartões microSD, a proteção do IP do usuário do projeto pode ser feita somente pelo mecanismo de proteção com senha da POU no Connected Components Workbench (Developer Edition) e NÃO pelo recurso Controller Lock.

IMPORTANTE Se o ajuste do parâmetro Load Always estiver habilitada e a energia for perdida ao restaurar um projeto do cartão microSD, o controlador tentará carregar o projeto usando o diretório e o nome do projeto padrão depois da restauração da energia. Se o seu projeto não estiver usando o nome e o diretório padrão, a operação falhará e ocorrerá uma falha ou o projeto incorreto será carregado.

O nome de projeto padrão é o nome do controlador, por exemplo "Micro850", e o diretório padrão é "Micro850\USERPRJ".

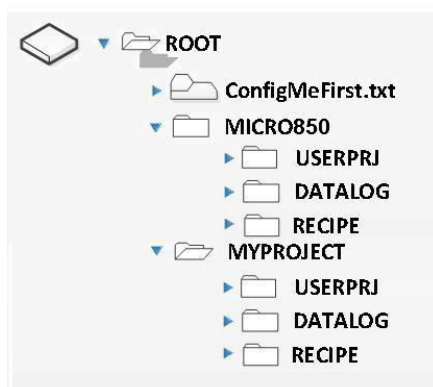
Se você alterar o nome do controlador do padrão, deverá configurar o UPD no arquivo ConfigMeFirst.txt.

O cartão microSD armazena a senha do controlador no formato encriptado. Quando a senha é diferente, o conteúdo do cartão microSD card não é restaurado no controlador.

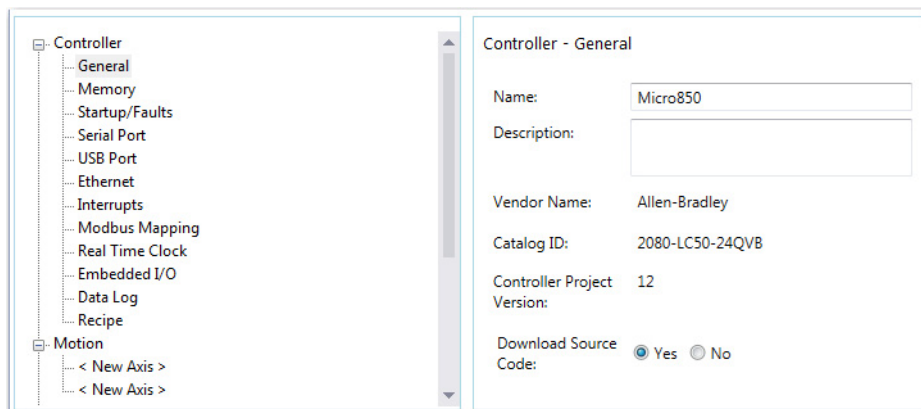
O backup e a restauração podem ser configurados para serem acionados da seguinte maneira:

Método	Backup	Restaurar
Online com o software Connected Components Workbench	Sim	Sim
Configuração do projeto no cartão de memória na energização	Não	Opções Load Always e/ou Load on Memory Error options
ConfigMeFirst.txt na energização	Sim (pelo comando [BKD])	Sim (Pelo comando [RSD])

Backup e restauração da estrutura do diretório



Ao fazer o backup de um projeto do usuário, é criado um subdiretório chamado <Micro800>\USERPRJ no cartão microSD. O nome da pasta assume o nome do projeto especificado na página General no software Connected Components Workbench, que, por padrão, é o nome do controlador. No entanto, se o arquivo ConfigMeFirst.txt especificar um subdiretório diferente (por exemplo: MyProject), o backup do projeto será realizado nesse diretório. Consulte [Regras gerais de configuração no ConfigMeFirst.txt na página 234](#).



A restauração do projeto é feita a partir do subdiretório especificado no arquivo ConfigMeFirst.txt ou da pasta padrão < Micro800>/USERPRJ, se nada for especificado no arquivo ConfigMeFirst.txt. O usuário precisa garantir que o diretório seja preenchido com o conteúdo correto antes da restauração.

O arquivo ConfigMeFirst.txt é um arquivo de configuração armazenado no cartão microSD que o usuário pode, opcionalmente, criar para personalizar os diretórios de backup, restauração, receita e registro de dados. As seções a seguir incluem informações sobre como configurar o ConfigMeFirst.txt corretamente.

IMPORTANTE O controlador Micro800 informa uma falha grave quando o backup do projeto não é bem-sucedido devido ao tamanho do cartão de memória ser excedido.

Ajustes de parâmetros de energização no ConfigMeFirst.txt

Na energização, o controlador lê e executa as definições de configuração descritas no arquivo ConfigMeFirst.txt. No entanto, a configuração do UPD ocorre quando o cartão microSD é inserido. As definições de configuração do arquivo ConfigMeFirst.txt são mostradas na tabela a seguir.

Definições de configuração do ConfigMeFirst.txt

Ajuste de parâmetro	Entra em vigor em...	Descrição
Configurações da atualização do firmware		
[FWFILE]	energizar	Local do caminho de arquivo da versão do firmware no cartão microSD. O local padrão está no seguinte formato: firmware\<catalog number>\<filename of firmware>
[FWDOWN]	energizar	Define se deve atualizar ou rebaixar o firmware do controlador da versão atual. 0 = Atualizar o firmware; 1 = Rebaixar o firmware IMPORTANTE: A atualização do firmware acontecerá se o ajuste do parâmetro [FWFILE] apontar para uma versão mais nova do arquivo do firmware em comparação com o firmware atual no controlador, independentemente do ajuste do parâmetro [FWDOWN].
Configurações do controlador		
[PM]	energizar	Energização e mudança para o modo de programa.
[CF]	energizar	Energização e tentativa de limpar uma falha.

Definições de configuração do ConfigMeFirst.txt (Continuação)

Ajuste de parâmetro	Entra em vigor em...	Descrição
Configurações de projeto		
[BKD = My Proj1]	energizar	Energização e salva o projeto do controlador no diretório de backup, My Proj1\USERPRJ. É necessário um ciclo de alimentação extra para primeiro limpar as falhas existentes usando a configuração [CF] ou outros meios.
[RSD = MyProj2]	energizar	Energização e leitura do projeto a partir do diretório de restauração MyProj2\USERPRJ no controlador. É necessário um ciclo de alimentação extra para primeiro limpar as falhas existentes usando a configuração [CF] ou outros meios. Este ajuste sobrescreve a carga UPD (ou seu padrão) sempre ou a carga na função de restauração do erro.
[UPD = My Proj]	Energização e inserção	Defina o nome do diretório do projeto do usuário para uso normal de backup e restauração (ou seja, por meio do software Connected Components Workbench, pelas configurações Load Always ou Load on Memory Error). Por exemplo, My Proj, durante a energização ou quando é inserido o cartão microSD. Este diretório também é usado pela função registro de dados e receita.
Configurações de rede		
[ESFD]	energizar	Ajustes de fábrica seriais incorporados. Energização e reversão da comunicação serial incorporada para os ajustes de fábrica.
[IPA = xxx.xxx.xxx.xxx]	energizar	Energização e definição do endereço IP como xxx (somente números).
[SNM = xxx.xxx.xxx.xxx]	energizar	Energização e definição da máscara de sub-rede como xxx (somente números).
[GWA = xxx.xxx.xxx.xxx]	energizar	Energização e definição do endereço de gateway como xxx (somente números).
Ajustes gerais		
[END]	energizar	Encerrar o ajuste de parâmetro. Esta configuração é sempre necessária, mesmo quando o arquivo ConfigMeFirst.txt não contém nenhuma outra Configuração. O LED SD apaga quando esta configuração não estiver presente.

IMPORTANTE Atualização flash de ajustes de parâmetros

Com o software Connected Components Workbench versão 12 ou posterior, você pode atualizar o controlador Micro800 a partir do cartão microSD além de usar ControlFLASH. Consulte [Atualização flash a partir do cartão microSD na página 267](#) para obter instruções.

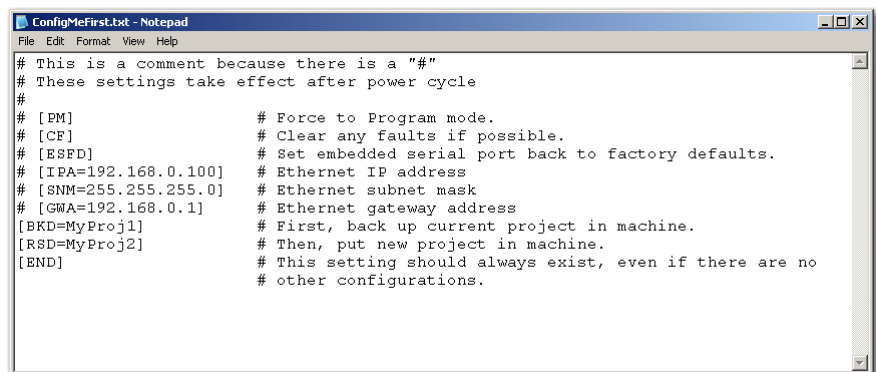
- Os ajustes [FWFILE], [FWBUS], [FWSLOT] e [FWDOWN] devem ser colocados no início do arquivo.

IMPORTANTE Configurações do diretório

- Se nenhum diretório for especificado no arquivo ConfigMeFirst.txt, o backup e a restauração ocorrerão no diretório de nome do controlador (<Micro800>/USERPRJ, por padrão).
- Se [UPD] for configurado no arquivo ConfigMeFirst.txt, então backup e a restauração ocorrerá no diretório [UPD] especificado.
- A configuração [BKD] é implementada mesmo quando o controlador é bloqueado ou a senha protegida.
- O diretório [BKD] é criado automaticamente, caso ainda não exista.

IMPORTANTE Ajustes de parâmetros de rede na energização

- [IPA], [SNM] e [GWA] seguem as regras de configuração IP gerais.
- [IPA], quando definido em ConfigMeFirst.txt, deve sempre ser configurado com um [SNM] válido e vice-versa.
- Quando a configuração opcional [GWA] é usada, certifique-se de que as configurações [IPA] e [SNM] estejam também presentes no ConfigMeFirst.txt.
- As configurações [ESFD], [IPA], [SNM] e [GWA] substituem as respectivas configurações de comunicação de restauração do projeto devido a [RSD], Load Always ou Load on Memory Error.

Exemplo de arquivo ConfigMeFirst.txt


```

# This is a comment because there is a "#"
# These settings take effect after power cycle
#
# [PM]                # Force to Program mode.
# [CF]                # Clear any faults if possible.
# [ESFD]              # Set embedded serial port back to factory defaults.
# [IPA=192.168.0.100] # Ethernet IP address
# [SNM=255.255.255.0] # Ethernet subnet mask
# [GWA=192.168.0.1]   # Ethernet gateway address
[BKD=MyProj1]         # First, back up current project in machine.
[RSD=MyProj2]         # Then, put new project in machine.
[END]                 # This setting should always exist, even if there are no
                     # other configurations.

```

Regras gerais de configuração no ConfigMeFirst.txt

- Todas as configurações deve estar em letra maiúscula e entre colchetes [].
- Cada linha deve conter apenas uma configuração.
- As configurações devem sempre aparecer primeiro em uma linha.
- Os comentários são iniciados pelo símbolo #.
- Nenhuma ação relacionada à configuração será executada quando a configuração não existe ou se um símbolo # aparecer antes da configuração (Exemplo, #[PM]).

Erros do ConfigMeFirst.txt

O LED de status do SD apaga quando o cartão microSD é inserido durante o modo de PROGRAMA ou de OPERAÇÃO (ou na energização) e o arquivo ConfigMeFirst.txt não puder ser lido ou estiver inválido. O arquivo ConfigMeFirst.txt será inválido quando tiver os seguintes erros:

- configuração não reconhecida (ou seja, as primeiras três regras de configuração não foram seguidas),
- o ajuste de parâmetro após o símbolo = é inválido, não existe ou está fora de faixa,
- a mesma configuração ocorre duas vezes ou mais,
- um ou mais caracteres sem configuração existe no mesmo colchete,
- há espaço entre os caracteres de configuração (Exemplo, [P M]), ou
- espaço entre o endereço IP, máscara de sub-rede e endereço de gateway (por exemplo, xxx. x xx.xxx.xxx)

- somente um dos ajustes do parâmetro de rede ([IPA], [SNM] ou [GWA]) é atribuído
- A configuração [END] não existe (mesmo que não existam outras configurações no arquivo de configuração).

O cartão microSD card torna-se não-utilizável até que o arquivo ConfigMeFirst.txt possa ser lido ou se os erros forem corrigidos.

Entregar atualizações de projetos aos clientes por e-mail

Uma vantagem de usar o recurso de backup e restauração do projeto é permitir que você forneça atualizações de projetos aos clientes por e-mail. Você pode fazer isso seguindo o exemplo mostrado abaixo.

Projeto de backup para cartão microSD

O primeiro passo é fazer backup do projeto do controlador no cartão microSD.

1. No software Connected Components Workbench, verifique se você fez o download do projeto atualizado para seu controlador.
2. Insira o cartão microSD no slot do cartão microSD.
3. Defina o controlador no modo de programa.
4. Na opção cartão de memória nos ajustes de parâmetro no seu controlador, clique em Backup do cartão de memória.

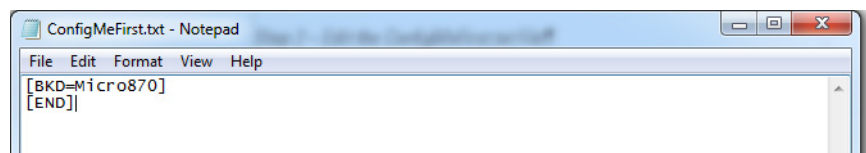
IMPORTANTE O botão backup para cartão de memória é ativado quando o controlador está no modo de programação e um cartão microSD está no slot para cartão microSD.

5. Após a conclusão do backup, clique em OK.

Os arquivos de imagem são armazenados no local padrão no cartão microSD <Micro800>\\USERPRJ. É desse local que o controlador carrega quando o parâmetro Load on power up estiver configurado para “Load Always” ou “Load on Error”.

Como alternativa, se você não quiser usar o software Connected Components Workbench para criar o backup do projeto, também poderá usar o arquivo ConfigMeFirst.txt.

Figura 19 - Exemplo de configuração para o backup do projeto

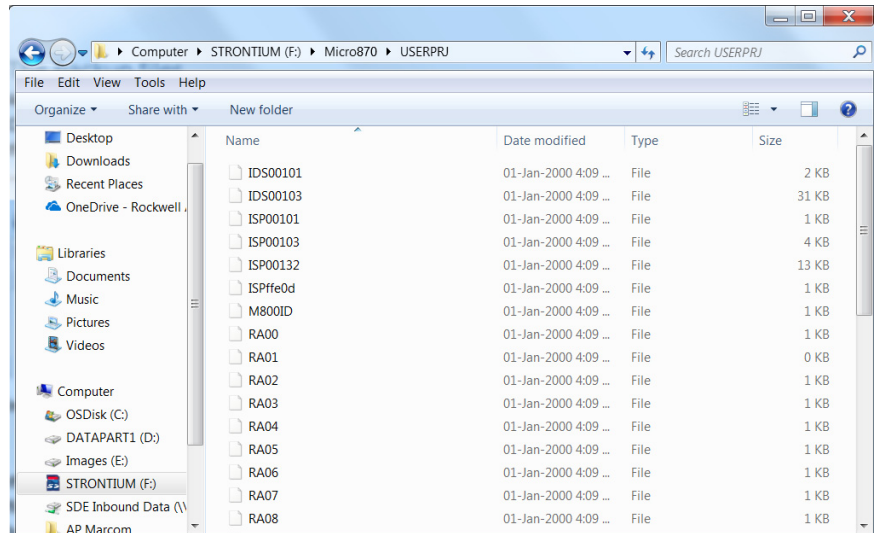


O arquivo ConfigMeFirst.txt também permite que você restaure a partir do backup, se você quiser configurar a configuração Carregar ao energizar em “Disable”.

Enviar arquivos de imagem por e-mail

O próximo passo é recuperar os arquivos de imagem do cartão microSD e enviá-los ao seu cliente por e-mail.

1. Remova o cartão microSD do controlador e leia-o pelo computador.
2. Navegue até o local onde os arquivos de imagem estão armazenados (padrão é <Micro800>\USERPRJ).



3. Use um programa de compactação para compactar esses arquivos de imagem e enviá-los para seu cliente por e-mail.

O cliente deve descompactar esses arquivos de imagem na raiz do diretório do cartão microSD e verificar se o local é idêntico ao original (o padrão é <Micro800>\USERPRJ).

Restaurar o projeto a partir do backup

O último passo é restaurar o projeto para o seu controlador a partir do cartão microSD. Existem dois métodos para restaurar o backup, dependendo da configuração do controlador.

Controlador existente - Load Always / Load on Error

Para este exemplo, a configuração Load on power up foi configurada para "Load Always". Isso significa que o controlador carrega o projeto do cartão de memória sempre que é ligado.

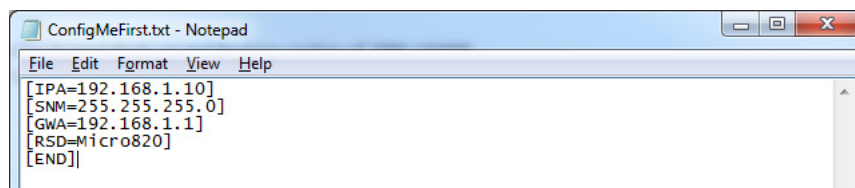
1. Insira o cartão microSD no slot do cartão microSD.
2. Desligue e ligue a alimentação do controlador.
3. Quando o LED SD exibe uma luz verde fixa, a restauração do projeto está concluída.

Esse método é usado para um controlador existente que foi configurado e você deseja atualizar o programa.

Novo controlador

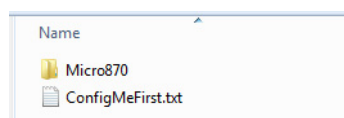
Se o seu controlador for novo, você poderá usar o arquivo ConfigMeFirst.txt para restaurar o backup do projeto.

Figura 20 - Exemplo de configuração para restauração de projeto



No exemplo mostrado acima, o arquivo ConfigMeFirst.txt configura o endereço IP, a máscara de sub-rede e o gateway do controlador e restaura o projeto do local especificado no cartão microSD.

O arquivo ConfigMeFirst.txt deve ser colocado na mesma raiz do diretório que a pasta de backup no cartão microSD.



1. Insira o cartão microSD no slot do cartão microSD.
2. Desligue e ligue a alimentação do controlador.
3. Quando o LED SD exibe uma luz verde fixa, a restauração do projeto está concluída.

Registro de dados

O recurso de registro de dados permite capturar variáveis globais e locais com registro de data e hora do controlador Micro800 no cartão microSD. Você pode recuperar os datasets gravados no cartão microSD card lendo o conteúdo do cartão microSD card através da leitora de cartão ou fazendo o upload através do software Connected Components Workbench.

É suportado um número máximo de 10 datasets por um programa Micro800. Cada dataset pode conter até 128 variáveis, com um máximo de quatro variáveis de grupos de dados por dataset. As variáveis de grupo podem ter no máximo 252 caracteres. Todos os datasets são gravados no mesmo arquivo. Para mais informações sobre como os registros de dados são armazenados no cartão microSD, consulte [Estrutura do diretório de registro de dados na página 238](#).

Você pode recuperar arquivos de registro de dados do cartão microSD usando uma leitora de cartão ou carregando os registros de dados por meio do software Connected Components Workbench.

IMPORTANTE Recomenda-se carregar os arquivos de registro de dados no modo PROGRAM para um desempenho ideal e para evitar conflito de acesso ao arquivo. Por exemplo, se a instrução de registro de dados estiver em execução, o software Connected Components Workbench não carregará o último arquivo de registro de dados.

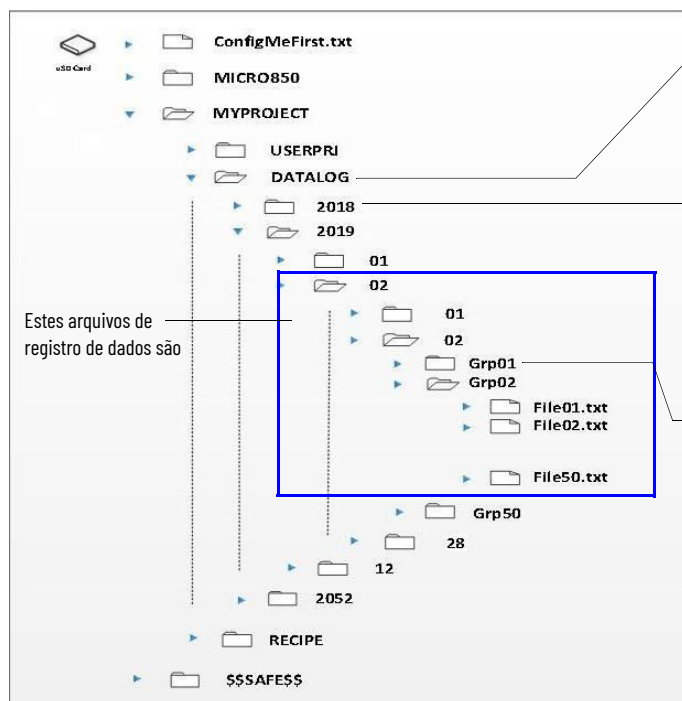
Consulte o exemplo do projeto de início rápido para começar a usar o recurso registro de dados em [Usar o recurso de registro de dados na página 247](#).

IMPORTANTE O tempo de execução de registro de dados depende da aplicação do usuário e sua complexidade. Os usuários são aconselhados a registrar dados no máximo a cada dois segundos para aplicações típicas. Observe que a manutenção leva pelo menos 5 ms por varredura de programa. Consulte [Execução de programa no Micro800 na página 137](#) para obter mais informações sobre varredura de programa e regras e sequência de execução.

Consulte também [Registro de dados – Carga útil de dados vs. tempo de desempenho na página 242](#).

IMPORTANTE Observe que em casos onde há a execução de blocos de funções RCP e DLG simultânea ou uploads/downloads/pesquisas, as atividades são colocadas em fila e manuseadas uma a uma pela varredura do programa. Você perceberá uma diminuição no desempenho nestes casos.

Estrutura do diretório de registro de dados



A pasta DATALOG é criada abaixo do diretório do projeto atual no cartão microSD. Neste exemplo, o diretório do projeto atual é MYPROJECT. Por padrão, o nome do diretório do projeto atual é obtido do nome do controlador do projeto baixado ou do ConfigMeFirst.txt. Consulte [Definições de configuração do ConfigMeFirst.txt na página 232](#).

Os subdiretórios também são criados seguindo o registro de data e hora RTC do controlador. Isto significa que se a data RTC no momento da execução do bloco de funções for 02 de fevereiro de 2019, é criada a subpasta 2019 abaixo do DATALOG. Abaixo da pasta 2019, é criada a subpasta 02 (que significa o mês de Fevereiro). Abaixo de 02, é criada outra subpasta 02, que corresponde ao dia atual.

Abaixo da pasta de trabalho atual, é criada a sub-pasta Grp01. Um máximo de 50 pastas Grpxxx pode ser gerado no cartão microSD card por dia.

Abaixo da pasta de trabalho Grpxxx atual, é criado o arquivo de registro de dados File01.txt. Uma vez que o arquivo atinge mais de 4 KB, outro arquivo, File02.txt, é criado automaticamente para armazenar dados. O tamanho do arquivo é mantido pequeno a fim de minimizar a perda de dados caso o cartão seja removido ou se houver um corte de energia inesperado.

Cada pasta Grpxx pode acomodar até 50 arquivos. Isto significa que, por exemplo, quando a pasta Grp01 já armazena 50 arquivos, uma nova pasta Grp02 é criada automaticamente para armazenar os próximos arquivos de registro de dados para esse dia. Esta pasta automática e a geração do arquivo continuam até que a pasta Grpxx atinja 50 para esse dia. Quando é inserido um cartão microSD, o bloco de funções DLG procura a última pasta Grpxx e o arquivo filexx.txt e continua a fazer o registro de dados com base nessas informações.

A tabela a seguir resume o desempenho do registro de dados nos controladores Micro800.

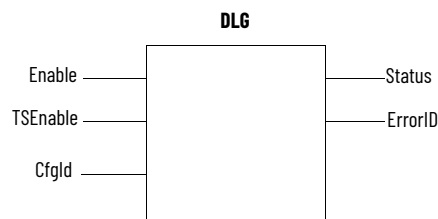
Especificações de registro de dados

Atributo	Valor	
Máximo de datasets	10	Todos os datasets são armazenados no mesmo arquivo.
Variáveis máximas por dataset	128	Configurado no software Connected Components Workbench.
Tamanho mínimo por arquivo	4 KB	
Arquivos máximos por pasta Grpxx ⁽¹⁾	50	Quando o diretório está cheio, um novo diretório é criado automaticamente no modo de operação.
Arquivos máximos (Filexx.txt) por dia	50	Quando o arquivo atinge o tamanho máximo, um novo arquivo é criado automaticamente no modo de operação.
Dados típicos por dia	10 MB	

(1) Quando o limite de registro de dados é atingido (ou seja, 50 pastas Grpxx por dia), é retornado um erro (ErrorID 3: DLG_ERR_DATAFILE_ACCESS).

Bloco de funções de registro de dados (DLG)

O bloco de funções de registro de dados permite que um programa do usuário grave valores globais de tempo de execução no arquivo de registro de dados no cartão microSD.



Parâmetros de entrada e saída do DLG

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição
permitir	ENTRADA	BOOL	Habilitação da função de escrita do registro de dados. Na borda de subida (ou seja, o valor de habilitação é acionado de baixo para alto), o bloco de funções executa. A pré-condição para execução é a de que a última operação seja concluída.
TSEnable	ENTRADA	BOOL	Flag de habilitação de armazenamento de data e registro de data e hora.
Cfgld	ENTRADA	USINT	Número (1 a 10) de dataset (DSET) configurado.
Status	SAÍDA	USINT	Status atual do bloco de funções de registro de dados.
ErrorID	SAÍDA	UDINT	Error ID se DLG Write falhar.

Status do bloco de funções DLG

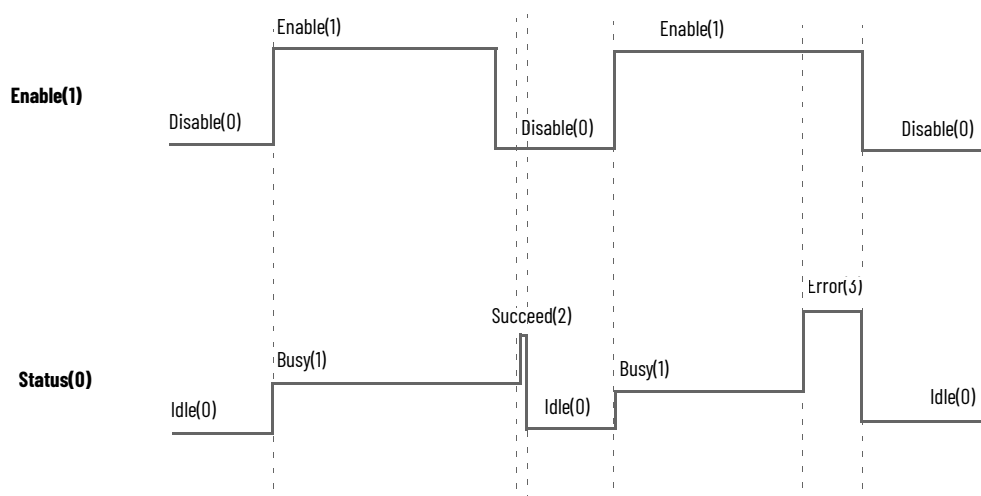
Código de status	Descrição
0	Status IDLE de registro de dados.
1	Status BUSY de registro de dados.
2	Status COMPLETE SUCCEED de registro de dados.
3	Status COMPLETE ERROR de registro de dados.

Erros do bloco de funções DLG

Código de status	Nome	Descrição
0	DLG_ERR_NONE	Sem erro.
1	DLG_ERR_NO_SDCARD	Não há cartão microSD.
2	DLG_ERR_RESERVED	Reservado.
3	DLG_ERR_DATAFILE_ACCESS	Erro ao acessar o arquivo de registro de dados no cartão microSD.
4	DLG_ERR_CFG_ABSENT	Falta o arquivo de configuração de registro de dados.
5	DLG_ERR_CFG_ID	Falta o ID de configuração no arquivo de configuração de registro de dados.
6	DLG_ERR_RESOURCE_BUSY	O mesmo ID de configuração é usado com outra chamada do bloco de funções de registro de dados ao mesmo tempo
7	DLG_ERR_CFG_FORMAT	O formato do arquivo de configuração de registro de dados está errado.
8	DLG_ERR_RTC	O relógio em tempo real é inválido.
9	DLG_ERR_UNKNOWN	Ocorreu um erro não especificado.

IMPORTANTE Erro de acesso do arquivo será retornado durante a execução do bloco de funções DLG quando o cartão está cheio.

Figura 21 - Diagrama de temporização do bloco de funções de registro de dados



IMPORTANTE Execução do bloco de funções de registro de dados

- Existem três estados possíveis para o bloco de funções de registro de dados: Idle, Busy e Complete (que inclui Complete with Succeed e Complete with Error).
- Para uma execução do bloco de funções de registro de dados, o status típico começa de Idle, depois Busy e termina com Complete. Para acionar outra execução de bloco de funções, o status precisa retornar primeiro para Inativo.
- O status Inativo muda para o status Ocupado somente quando o sinal de entrada Habilitar estiver na borda de subida. O status Complete entra no status Idle quando o sinal de entrada Enable for somente Disable status.
- Os parâmetros de entrada TSEnable e CfgId somente sofrem amostragem somente na borda de subida do parâmetro Enable input quando a execução de um novo bloco de funções começa. Durante a execução do bloco de funções, os parâmetros de entrada de TSEnable e CfgId são bloqueados e qualquer alteração é ignorada.
- Quando a execução é concluída, o status muda de Busy para Complete. Neste estágio, se a entrada Enable for False, o status muda para Idle após a indicação Complete por exatamente um tempo de varredura. Caso contrário, o status do bloco de funções é mantido como Complete até que a entrada Enable mude para False.
- O arquivo de registro de dados somente pode ser criado pelo bloco de instrução DLG. O software Connected Components Workbench somente pode carregar e excluir o arquivo de registro de dados.
- Há separadores entre cada variável de dados no arquivo de dados, que é definido durante a configuração no software Connected Components Workbench.
Consulte [Tipos de dados suportados para blocos de funções de registro de dados e de receita na página 241](#).
- Os valores de variáveis de dados são amostrados quando o bloco de funções de registro de dados está no estado Busy. No entanto, o arquivo de registro de dados é criado somente quando o bloco de funções de registro de dados está no estado Complete.

Tabela 46 - Tipos de dados suportados para blocos de funções de registro de dados e de receita

Tipo de dados	Descrição	Exemplo de formato no arquivo de registro de dados de saída
BOOL ⁽¹⁾	Lógica booleana com valores VERDADEIRO e FALSO	0: FALSO 1: VERDADEIRO)
SINT	Inteiro com sinal e de 8 bits	-128, 127
INT	Inteiro com sinal e de 16 bits	-32768, 32767
DINT	Inteiro com sinal e de 32 bits	-2147483648, 2147483647
LINT	Inteiro com sinal e de 64 bits	-9223372036854775808, 9223372036854775807
USINT(BYTE)	Inteiro sem sinal e de 8 bits	0, 255
UINT(WORD)	Inteiro sem sinal e de 16 bits	0, 65535
UDINT(DWORD)	Inteiro sem sinal e de 32 bits	0, 4294967295
ULINT(LWORD)	Inteiro sem sinal e de 64 bits	0, 18446744073709551615
REAL	Valor de ponto flutuante de 32 bits	-3,40282347E+38, +3,40282347E+38
LREAL	Valor de ponto flutuante de 64 bits	-1,7976931348623157E+308, +1,7976931348623157E+308

Tabela 46 - Tipos de dados suportados para blocos de funções de registro de dados e de receita (Continuação)

Tipo de dados	Descrição	Exemplo de formato no arquivo de registro de dados de saída
STRING ⁽²⁾	grupo de caracteres (1 byte por caractere)	"Rotation Speed"
DATE ⁽¹⁾	32 bits sem sinal valor inteiro	1234567 1234567 (As variáveis de data são armazenadas como palavras de 32 bits, um número positivo de segundos começando em 1970-01-01 à meia noite GMT).
TIME ⁽¹⁾	32 bits sem sinal valor inteiro	1234567 1234567 (As variáveis de hora são armazenadas como palavras de 32 bits, número positivo de milissegundos).

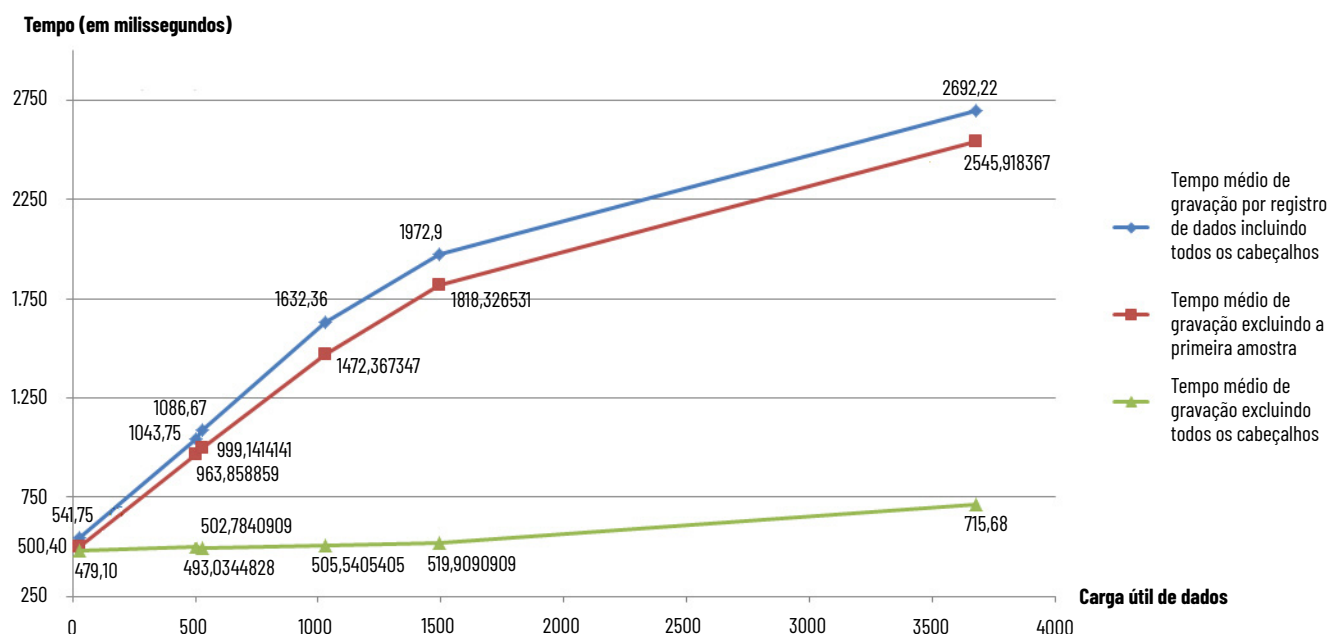
- (1) As variáveis de dados BOOL, DATE, TIME são apresentadas no formato digital decimal no cartão microSD card. Você tem a opção de converter este formato em um formato mais amigável. Por exemplo, use o bloco de funções ANY_TO_STRING para converter o tipo de dados BOOL (0, 1) em FALSE ou TRUE. De forma semelhante, faça o mesmo para os tipos de dados DATE e TIME. O tipo de dados DATE é apresentado no valor digital decimal diferencial entre a hora de base do sistema (1970/01/01,00:00:00) e o valor de data atual. Unidade em milissegundos. Time deve ser um valor de tempo absoluto. Unit é em segundos.
- (2) As variáveis de dados de grupo são apresentadas entre aspas no arquivo de registro de dados. O exemplo abaixo mostra DSET1 usando variáveis de grupo e DSET2 usando números inteiros.

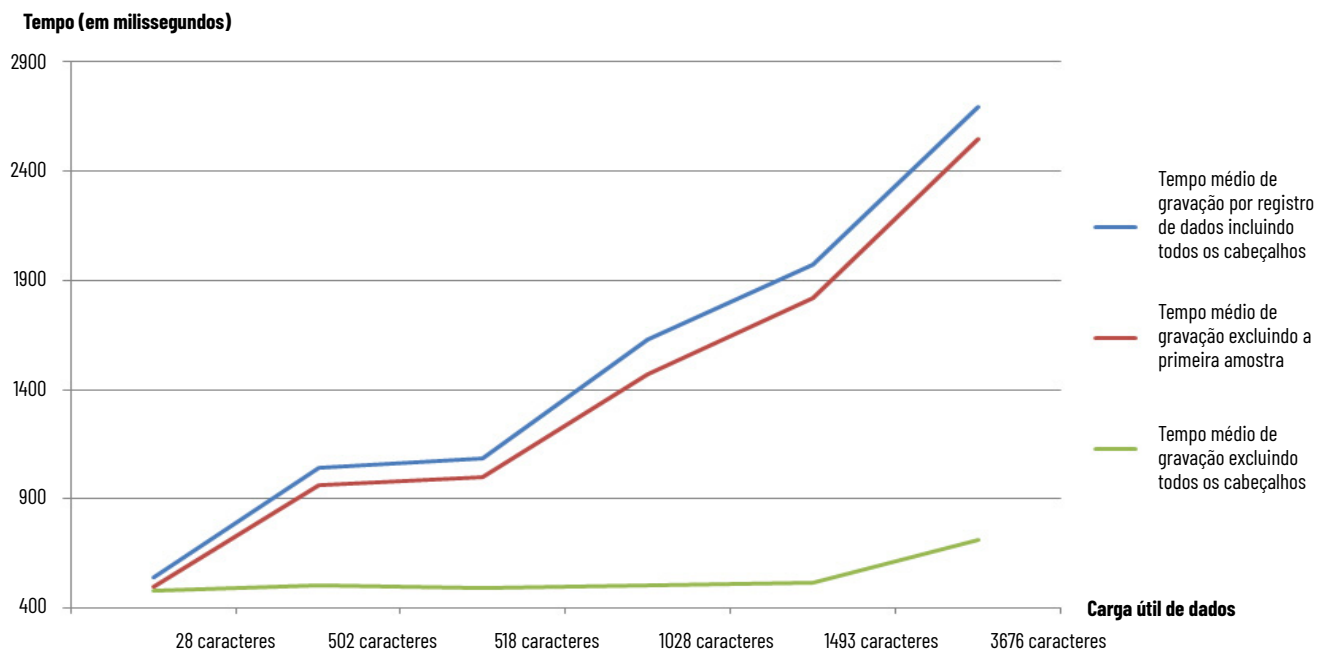
```
DSET1, "Temperature", "Humidity", "Pressure"
DSET2, 30, 50, 125
```

Desempenho de registro de dados

Registro de dados – Carga útil de dados vs. tempo de desempenho

Parâmetro	Número de caracteres					
	28	502	518	1028	1493	3676
Tempo médio de gravação por arquivo de registro de dados incluindo todos os cabeçalhos	541,77 ms	1043,75 ms	1086,67 ms	1632,36 ms	1972,9 ms	2696,22 ms
Tempo médio de gravação excluindo a primeira amostra	500,40 ms	963,86 ms	999,14 ms	1472,36 ms	1818,33 ms	2545,92 ms
Tempo médio de gravação excluindo todos os cabeçalhos	479,10 ms	493,03 ms	502,78 ms	505,54 ms	519,91 ms	715,68 ms





Receita

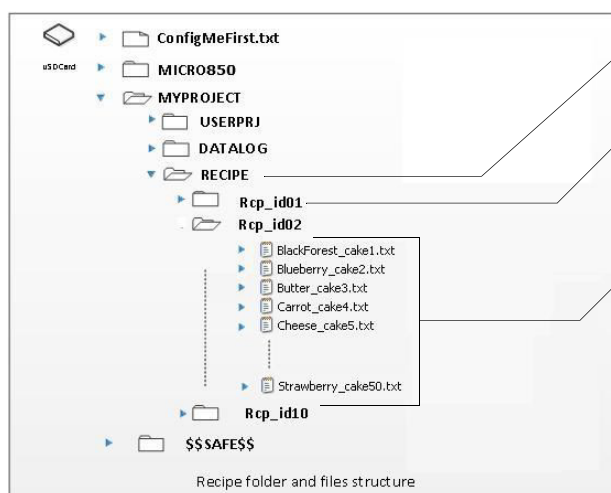
Os controladores Micro800 suportam recurso Receita e permitem que os usuários armazenem e carreguem uma lista de dados de e/ou para os arquivos de dados de receita usando a instrução RCP. Eles também permitem que os usuários baixem, carreguem e excluam dados de receita no cartão microSD por meio do software Connected Components Workbench.

É suportado um número máximo de 10 conjuntos de receitas por um programa Micro800. Cada receita pode conter até 128 variáveis, com um máximo de quatro variáveis de grupos de dados por receita. As variáveis de grupo podem ter no máximo 252 caracteres. Variações de receitas são armazenadas em arquivos separados com nomes de arquivos exclusivos. Para mais informações sobre como as receitas são armazenadas no cartão microSD, consulte [Estrutura do diretório de receita na página 244](#).

Tabela 47 - Especificações de receita

Atributo	Valor	
Número máximo de conjuntos de receitas	10	Os conjuntos de receitas são armazenados em 10 diretórios (Rcp_Id01 a Rcp_Id10) com um número máximo de 50 arquivos de receita em cada diretório.
Número máximo de receitas em cada conjunto	50	
Número máximo de variáveis por receita	128	Configurado no software Connected Components Workbench.
Bytes máximos por arquivo de receita	4 KB	

Estrutura do diretório de receita



Na primeira execução, o RCP cria uma pasta RECIPE no diretório do projeto atual no cartão microSD card.

Ele também cria 10 subdiretórios para cada conjunto de receitas com um nome seguindo o valor de entrada CfgID (1 a 10). Se o valor de CfgID for 1, será criada a subpasta Rcp_Id01.

Os arquivos de receita são então criados/gravados na pasta, com os nomes do arquivo que correspondem ao valor de entrada do parâmetro RcpName do bloco de funções RCP, conforme configurado no software Connected Components Workbench. Cada conjunto de receitas pode conter até 50 arquivos de receita ou variações delas. Os nomes de arquivos para arquivos de receita não devem exceder 30 caracteres.

Configuração e recuperação de receita

Você pode recuperar arquivos de receita a partir do cartão microSD usando uma leitora de cartão ou carregando e baixando os conjuntos de receita por meio do software Connected Components Workbench.

Bloco de funções de receita (RCP)

O bloco de funções RCP permite que um programa do usuário leia os valores de variável a partir de um arquivo de dados de receita existente, que está na pasta de receita do cartão microSD e atualizar os valores de variáveis globais e locais no controlador em tempo de execução. O bloco de funções RCP permite também que o programa do usuário grave valores de variável de tempo de execução global ou local a partir de um controlador menor no arquivo de dados de receita no cartão microSD.

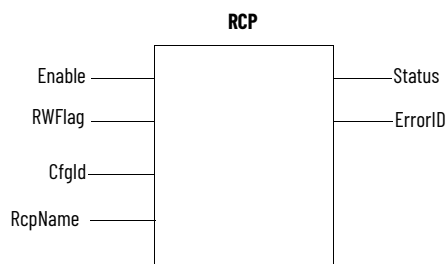


Tabela 48 - Parâmetros de entrada e saída RCP

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição
permitir	ENTRADA	BOOL	Habilitação da função de leitura/gravação de receita. Se a borda de subida (a habilitação é acionada de “baixo” para “alto”), inicia o bloco de funções de receita e a pré-condição é a de que a última operação esteja concluída.
RWFlag	ENTRADA	BOOL	TRUE: A receita grava as variáveis de dados nos arquivos de receita no cartão microSD. FALSE: A receita lê as variáveis de dados salvos do cartão microSD card e atualiza estas variáveis de acordo.
CfgId	ENTRADA	USINT	Número do conjunto de receita (1 a 10).
RcpName	ENTRADA	STRING	Nome do arquivo de dados de receita (máximo 30 caracteres).
Status	SAÍDA	USINT	Estado atual do bloco de funções de receita.
ErrorID	SAÍDA	UDINT	Informações de identificação do erro detalhado se a leitura/gravação RCP falhar.

Tabela 49 - Status do bloco de funções RCP

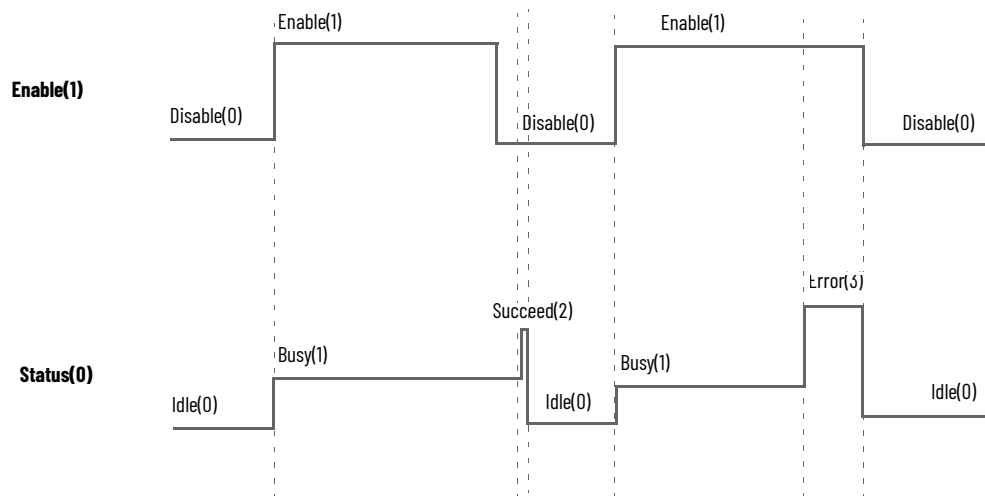
Código de status	Descrição
0	Status inativo da receita.
1	Status ocupado da receita.
2	Status bem-sucedido de conclusão da receita.
3	Status de erro na conclusão da receita.

Tabela 50 - Erros do bloco de funções RCP

ID do erro	Nome do erro	Descrição
0	RCP_ERR_NONE	Sem erro.
1	RCP_ERR_NO_SDCARD	Não há cartão microSD.
2	RCP_ERR_DATAFILE_FULL	Os arquivos de receita excedem o número máximo de arquivos por pasta de conjunto de receita.
3	RCP_ERR_DATAFILE_ACCESS	Erro ao acessar o arquivo de dados de receita no cartão microSD card.
4	RCP_ERR_CFG_ABSENT	Falta o arquivo de configuração de receita.
5	RCP_ERR_CFG_ID	Falta o Configure ID no arquivo de configuração de receita.
6	RCP_ERR_RESOURCE_BUSY	O recurso da operação de receita vinculado a este Recipe ID é usado por outra operação de bloco de funções.
7	RCP_ERR_CFG_FORMAT	O formato do arquivo de configuração de receita é inválido.
8	RCP_ERR_RESERVED	Reservado.
9	RCP_ERR_UNKNOWN	Ocorreu um erro não especificado.
10	RCP_ERR_DATAFILE_NAME	O nome do arquivo de dados de receita é inválido.
11	RCP_ERR_DATAFOLDER_INVALID	A pasta de dataset de receita é inválida.
12	RCP_ERR_DATAFILE_ABSENT	Falta o arquivo de dados de receita.
13	RCP_ERR_DATAFILE_FORMAT	Conteúdo errado no arquivo de dados de receita.
14	RCP_ERR_DATAFILE_SIZE	O tamanho do arquivo de dados de receita é muito grande (>4K).

IMPORTANTE Erro de acesso do arquivo será retornado durante a execução do bloco de funções RCP quando o cartão está cheio.

Figura 22 – Diagrama de temporização do bloco de funções de receita



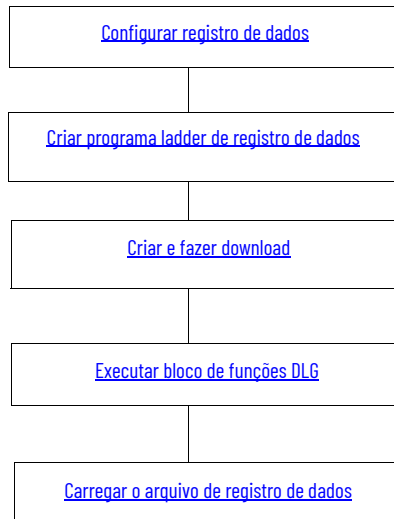
IMPORTANTE Execução do bloco de funções RCP

- Há três estados possíveis para o bloco de funções de receita: Idle, Busy, Complete (Complete with Succeed e Complete with Error)
- Para uma execução de bloco de funções de receita, o status típico começa de Idle, depois Busy e termina com Complete. Para acionar outra execução de bloco de funções, o status precisa retornar primeiro para Inativo.
- O status Inativo muda para o status Ocupado somente quando o sinal de entrada Habilitar estiver na borda de subida. O status Complete entra no status Idle quando o sinal de entrada Enable for Disable status.
- Os parâmetros de entrada RWFlag, CfgId e RcpName somente são amostrados na borda de subida do parâmetro Enable input quando começa a execução de um novo bloco de funções. Durante a execução do bloco de funções, os parâmetros de entrada de RWFlag, CfgId e RcpName são bloqueados e qualquer alteração é ignorada.
- Quando a execução do bloco de funções é concluída, o status do bloco de funções muda de Busy para Complete. Neste estágio, se a entrada Enable for False, o status do bloco de funções muda para Idle após permanecer na indicação Complete por exatamente um tempo de varredura. Caso contrário, o status do bloco de funções é mantido como Complete até que a entrada Enable mude para False.
- O nome do arquivo do bloco de funções de receita é compatível com no máximo 30 bytes de comprimento e somente letras maiúsculas e minúsculas Aa a Zz, números 0 a 9 e sublinhado (_).
- O parâmetro de entrada RcpName não permite extensão de arquivo (por exemplo, .txt) adicionada ao seu valor. O arquivo de dados de receita é gravado no cartão microSD card com a extensão .txt.
- Há separadores entre cada variável de dados no arquivo de dados de receita, que é definido durante a configuração no software Connected Components Workbench. Os caracteres redundantes de tabulação, espaço, retorno de carro e alimentação de linha são estritamente proibidos. Consulte [Tipos de dados suportados para blocos de funções de registro de dados e de receita na página 241](#).
- Aspas não são permitidas em um grupo no arquivo de receita.

Projetos de início rápido para blocos de funções de registro de dados e de receita

Os exemplos de projetos de início rápido a seguir oferecem instruções passo a passo sobre como usar os blocos de funções de registro de dados e de receita no software Connected Components Workbench para gerar e gerenciar seus arquivos de receita e de registros de dados.

Usar o recurso de registro de dados

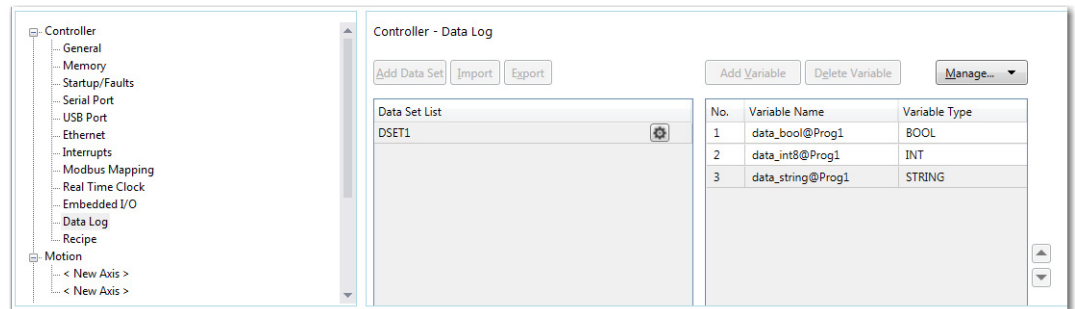


Configurar registro de dados

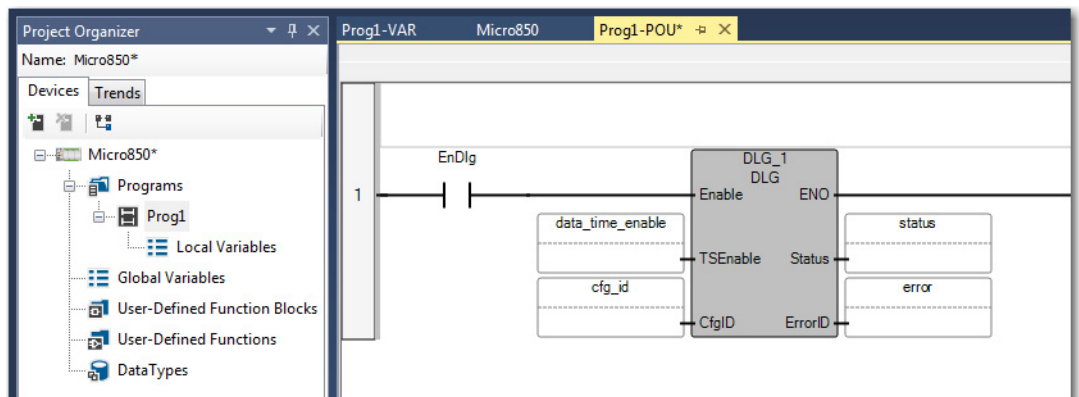
1. No software Connected Components Workbench, vá até o painel Properties para configurar seu registro de dados.
2. Selecione Data Log. Clique em Add Dataset para adicionar um dataset. Observe que cada dataset será armazenado no mesmo arquivo. Você pode adicionar até 10 datasets por configuração.
3. Clique em Add Variable para adicionar variáveis ao dataset. Você pode adicionar até 128 variáveis a cada dataset. Para este exemplo de projeto de início rápido, adicione as seguintes variáveis que foram criadas anteriormente no dataset 1.

Variáveis locais

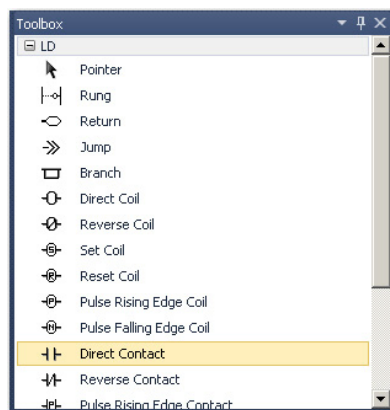
Nome da variável	Tipo de dados
data_bool	BOOL
data_int8	INT
data_string	STRING



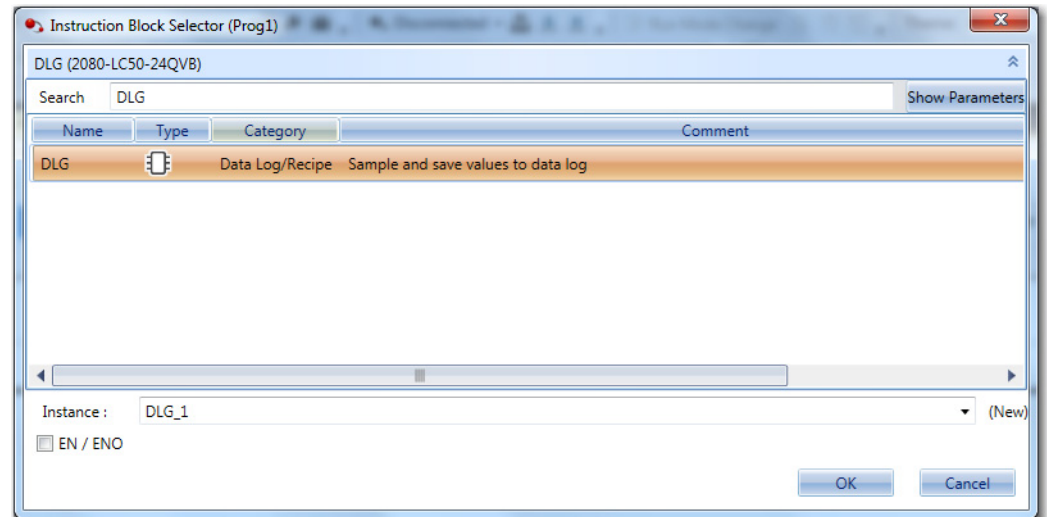
Criar programa ladder de registro de dados



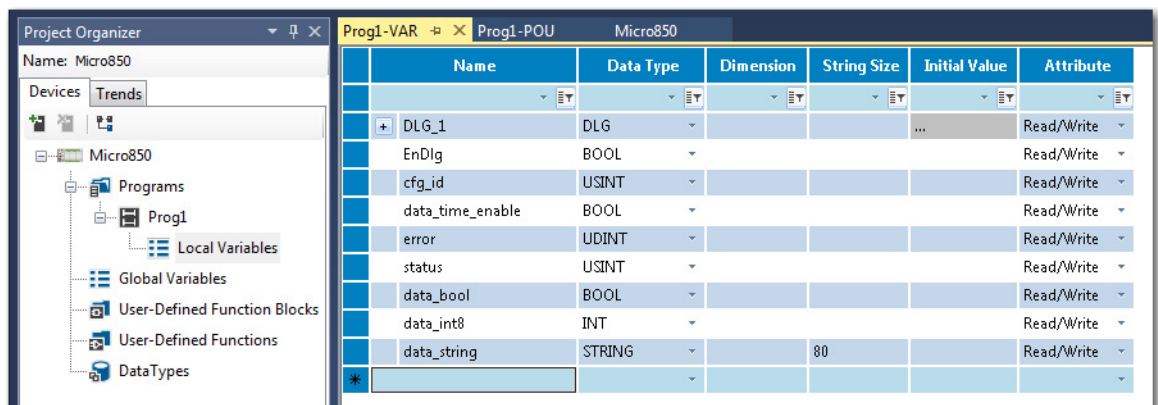
1. Execute o software Connected Components Workbench. Criar um programa do usuário para seu controlador Micro820.
2. Clique com o botão direito do mouse em Programs. Selecione Adicionar novo LD: diagrama ladder. Nomeie o programa (por exemplo, Prog1).
3. A partir de Toolbox, clique duas vezes em Direct Contact para adicioná-lo à linha de programa.



4. A partir de caixa de ferramentas, clique duas vezes em Block para adicioná-lo à linha.
5. Na janela Block Selector exibida, digite DLG para filtrar o bloco de funções DLG da lista de bloco de funções disponíveis. Clique em OK.



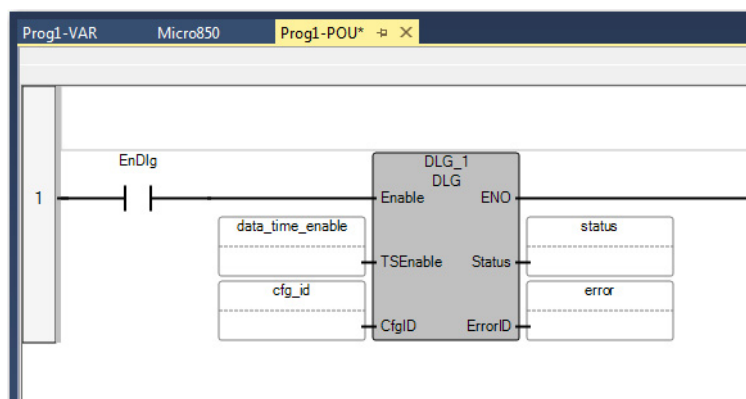
6. Crie as seguintes variáveis locais para seu projeto.



Variáveis locais

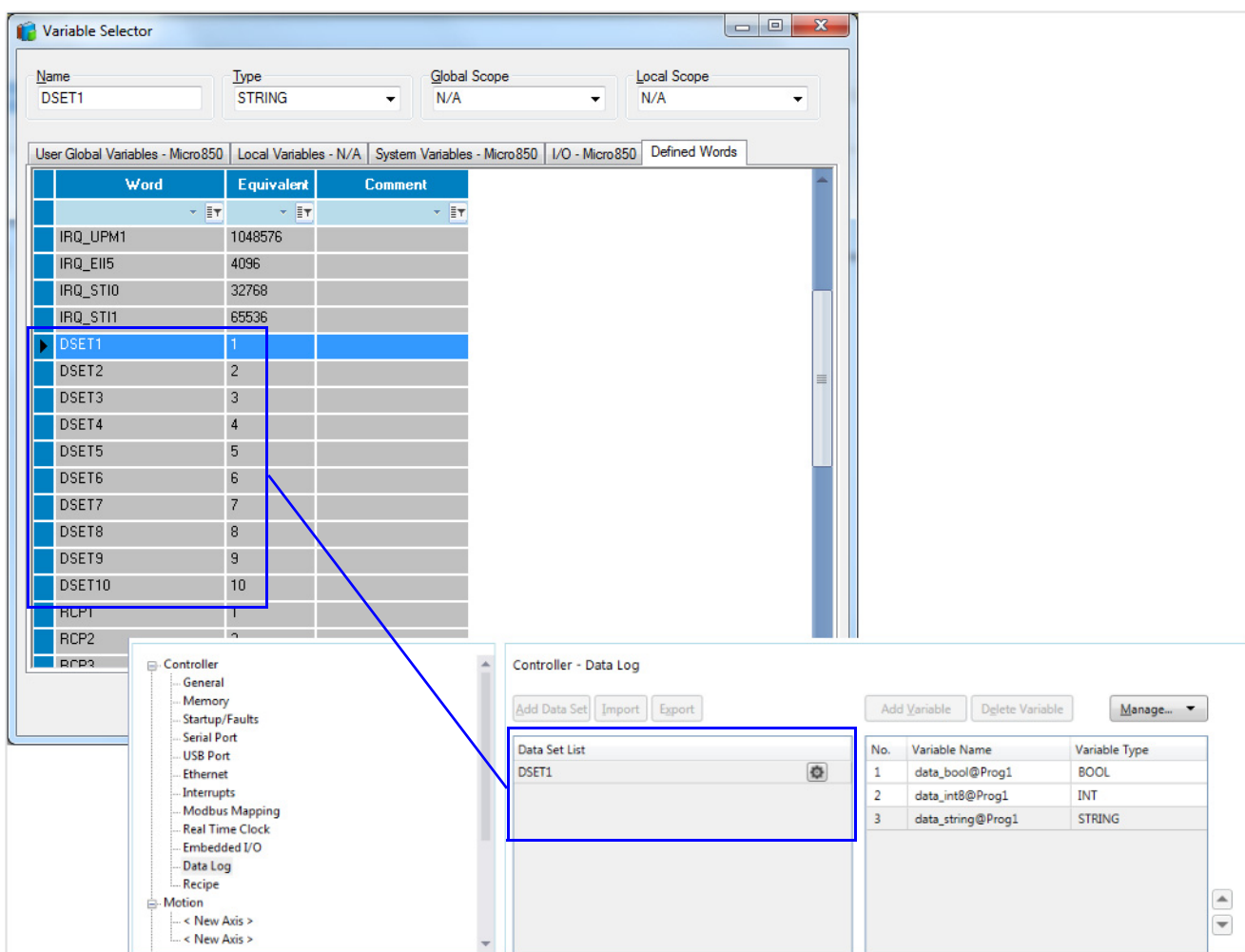
Nome da variável	Tipo de dados
EnDlg	BOOL
cfg_id	USINT
data_time_enable	BOOL
erro	UDINT
status	USINT
data_bool	BOOL
data_int8	INT
data_string	STRING

7. Atribua as variáveis aos parâmetros de entrada e saída DLG como segue:



Observação: para o parâmetro de entrada CfgID, você pode escolher uma variável predefinida escolhendo entre as Defined Words no software Connected Components Workbench. Para isso, clique na caixa de entrada CfgID. Na janela Variable Selector exibida, clique na guia Defined Words e selecione as palavras da lista de palavras. Por exemplo, DSET1 que corresponde a DSET1 na configuração de sua receita. Consulte a [Figura 23](#).

Figura 23 - Escolha uma variável predefinida

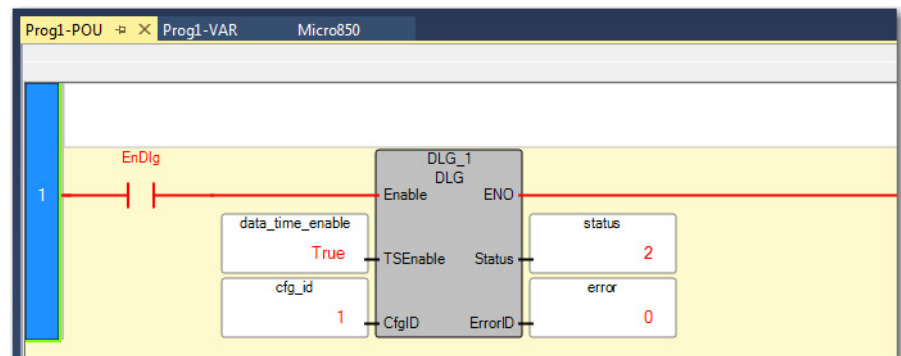


Criar e fazer download

Após configurar as propriedades de registro de dados, crie o programa e faça o download no controlador.

Executar bloco de funções DLG

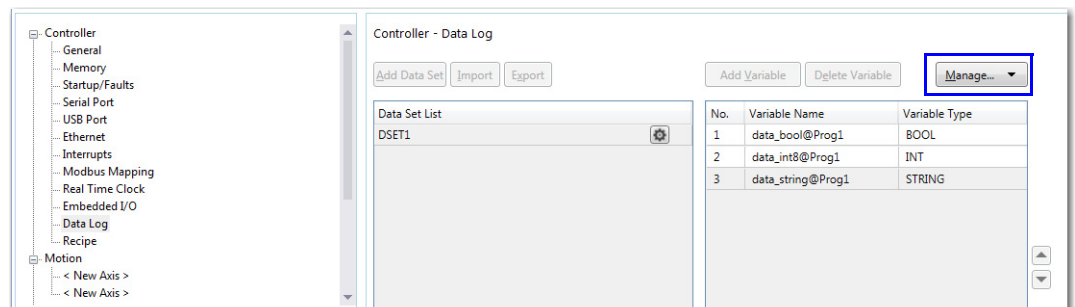
Execute o bloco de funções DLG. Observe que a saída Status vai de 0 (inativo) para 1 (ativo) e 2 (concluído).



Carregar o arquivo de registro de dados

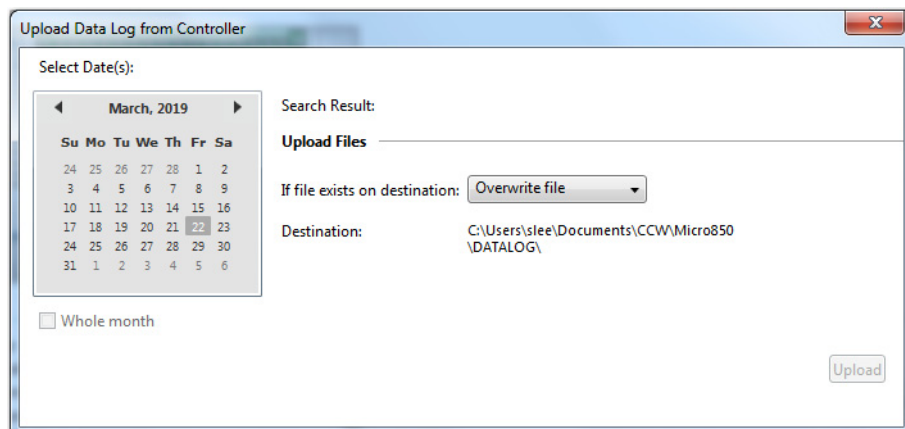
Você pode recuperar arquivos de registro de dados do cartão microSD usando uma leitora de cartão ou carregando os registros de dados por meio do software Connected Components Workbench.

1. Para usar o recurso de Upload, vá para a seção Properties de seu projeto no software Connected Components Workbench.
2. Selecione Data Log. Clique em Manage e escolha Upload.



IMPORTANTE O botão Manage não está disponível no modo DEBUG. É necessário parar o modo DEBUG para usar o botão Manage para carregar os arquivos de registro de dados. Recomenda-se carregar os arquivos de registro de dados no modo de programa por motivos de desempenho e de bloqueio de arquivo.

3. A partir da janela Upload que é exibida, selecione a data dos arquivos de registro de dados que deseja carregar. É possível carregar registros de dados do mês inteiro clicando no botão Whole Month.

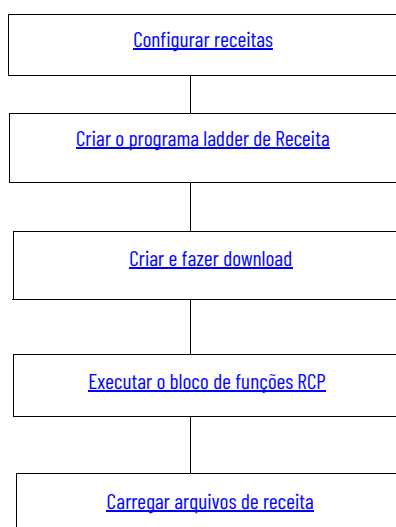


4. Se o arquivo já existir na sua pasta de destino, escolha entre Overwrite file, Skip file ou Preserve both files.
5. Clique em Upload. A barra de progresso informa se o carregamento foi bem-sucedido ou não.

IMPORTANTE Não remova o cartão microSD do slot enquanto os dados estão sendo gravados ou recuperados do cartão. As operações de gravação e de leitura em andamento são indicadas pelo LED de status do SD piscando.

IMPORTANTE Para uma melhor gestão do arquivo de registro de dados, é possível usar uma ferramenta de terceiros ou DOS CMD para mesclar todos os arquivos de registro de dados em um único arquivo e importar como um arquivo CSV em Excel.

Use o recurso Receita

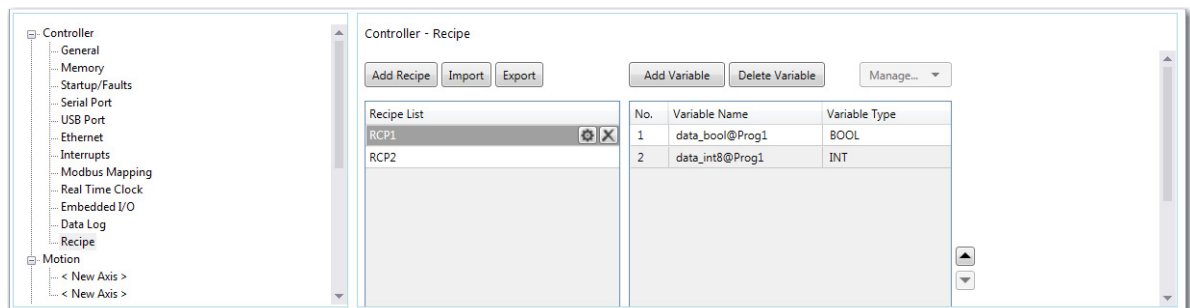


Configurar receitas

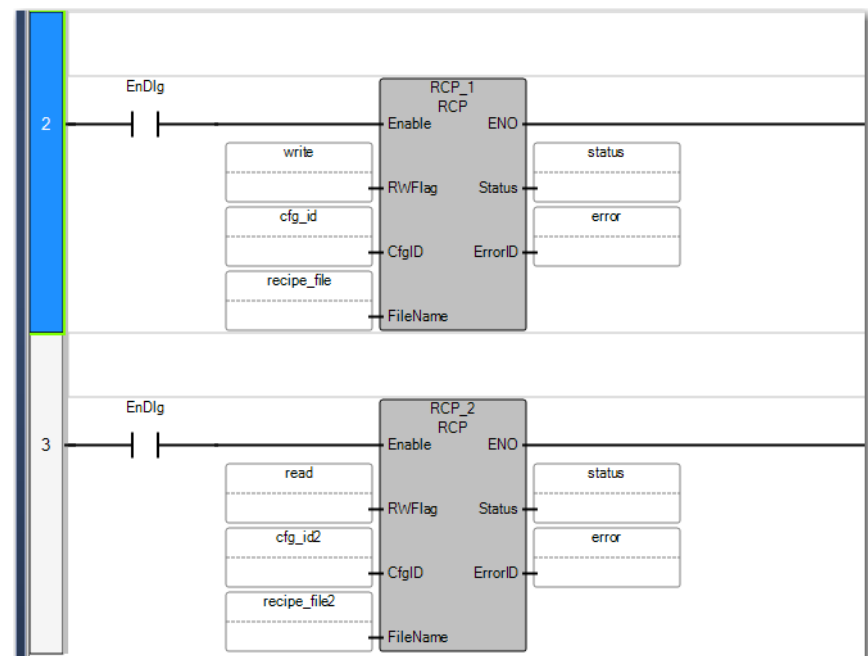
1. No software Connected Components Workbench, vá até o painel Properties para configurar a receita.
2. Selecione Recipe. Clique em Add Recipe para adicionar uma receita. Observe que cada receita será armazenada em arquivos separados. Você pode adicionar até 10 receitas por configuração.
3. Clique no botão Add Variable para adicionar variáveis à receita. Você pode adicionar até 128 variáveis a cada receita. Para este exemplo de projeto de início rápido, adicione as seguintes variáveis criadas anteriormente em RCP 1:

Variáveis locais

Nome da variável	Tipo de dados
data_bool	BOOL
data_int8	INT

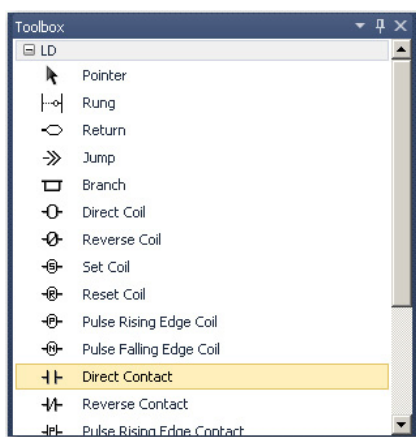


Criar o programa ladder de Receita

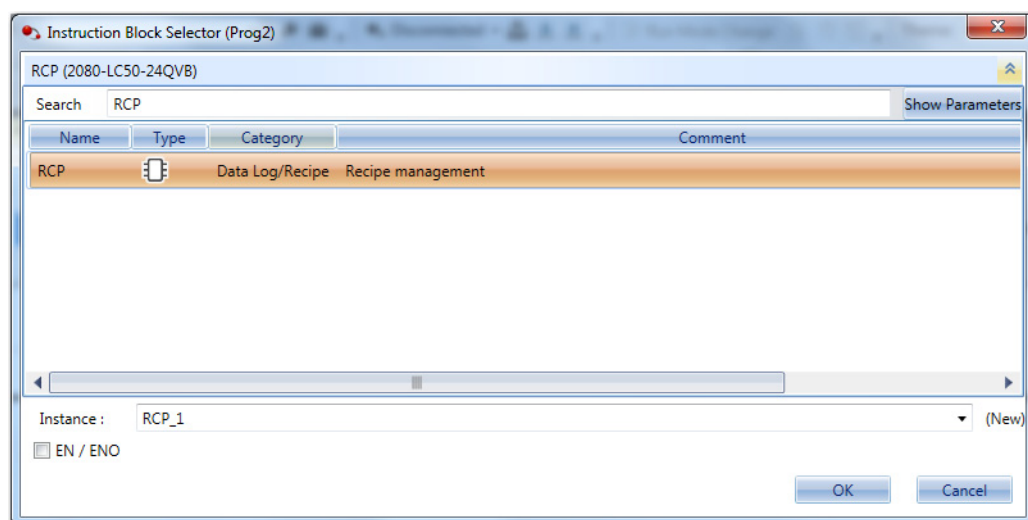


1. Execute o software Connected Components Workbench. Criar um programa do usuário para seu controlador Micro820.
2. Clique com o botão direito do mouse em Programs. Selecione Adicionar novo LD: diagrama ladder. Nomeie o programa (por exemplo, Prog1).

- Em Toolbox, clique duas vezes em Direct Contact para adicioná-lo à primeira linha de programa.



- A partir de caixa de ferramentas, clique duas vezes em Block para adicioná-lo à linha.
- Na janela Block Selector exibida, digite RCP para filtrar o bloco de funções Recipe da lista de bloco de funções disponíveis. Clique em OK.



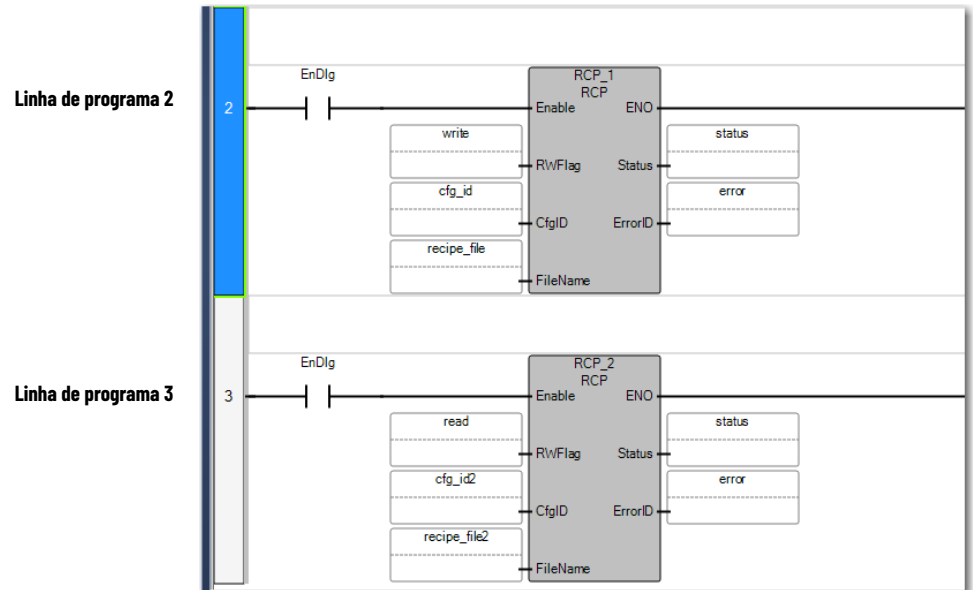
- A partir da Toolbox, clique duas vezes na linha para adicionar outra linha.
- Adicione um contato direto e um bloco de funções RCP a esta segunda linha seguindo as etapas 3 a 5.
- Crie as seguintes variáveis locais para seu programa, além das que já foram criadas para o registro de dados.

recipe_file	STRING	80	"MyFirstRecipe"	Read/Write
recipe_file2	STRING	80	"MySecondRecipe"	Read/Write
cfg_id2	USINT	2		Read/Write
read	BOOL		FALSE	Read/Write
write	BOOL		TRUE	Read/Write
RCP_1	RCP		...	Read/Write
RCP_2	RCP		...	Read/Write

Variáveis locais

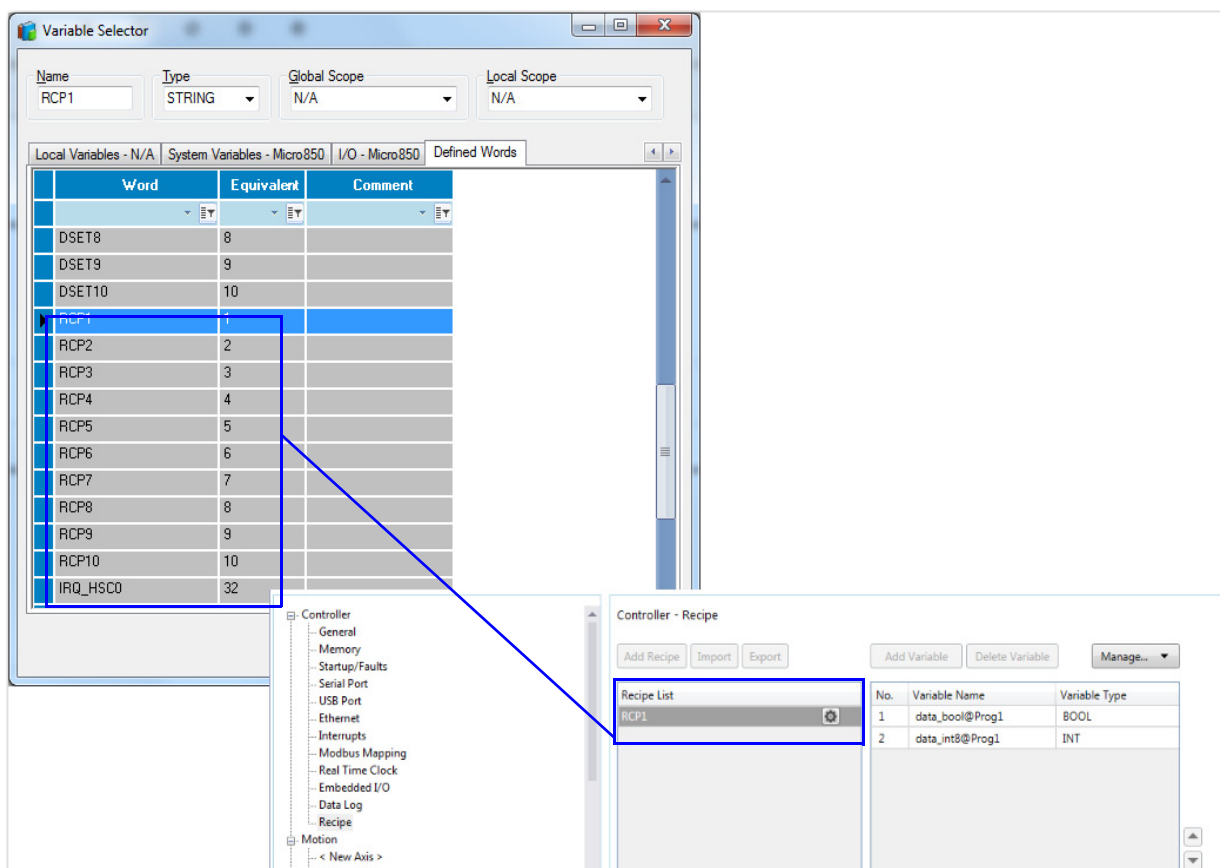
Nome da variável	Tipo de dados
recipe_file	STRING
recipe_file2	STRING
cfg_id2	USINT
ler	BOOL
write	BOOL

9. Atribua as variáveis aos parâmetros de entrada e saída RCP como segue:



Observação: para o parâmetro de entrada CfgID, você pode escolher uma variável predefinida escolhendo entre as Defined Words no software Connected Components Workbench. Para isso, clique na caixa de entrada CfgID. Na janela Variable Selector exibida, clique na guia Defined Words e selecione as palavras da lista de palavras. Por exemplo, RCP1 que corresponde a RCP1 em sua configuração de receita. Consulte a [Figura 24](#).

Figura 24 - Escolha uma variável predefinida

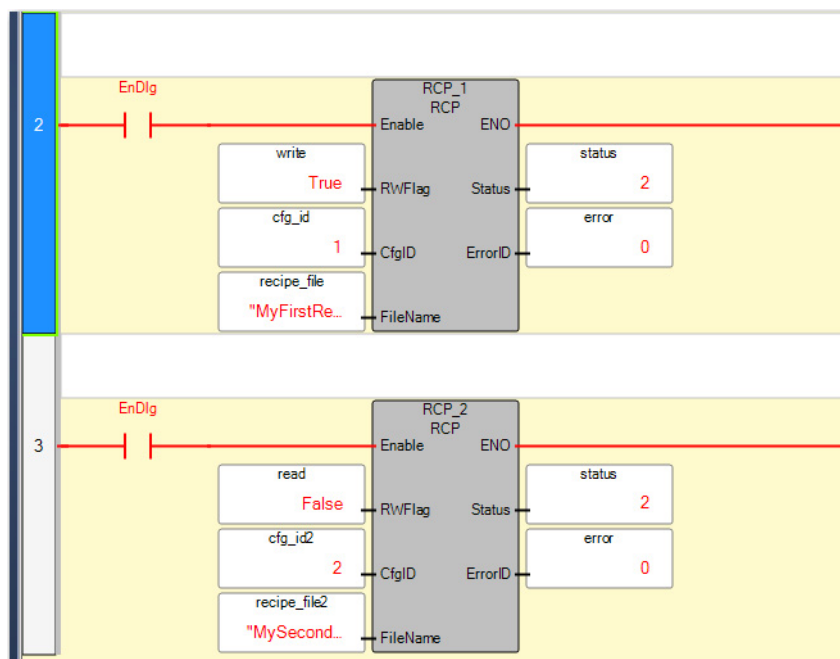


Criar e fazer download

Após configurar a Receita, crie o programa e faça o download no controlador.

Executar o bloco de funções RCP

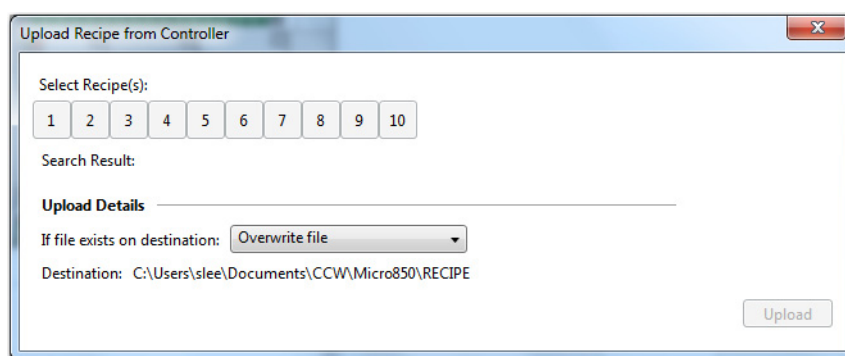
Execute o bloco de funções RCP. Observe que a saída Status vai de 0 (inativo) para 1 (ativo) e 2 (concluído).



Carregar arquivos de receita

Você pode recuperar arquivos de receita a partir do cartão microSD usando uma leitora de cartão ou carregando os arquivos de receita por meio do software Connected Components Workbench..

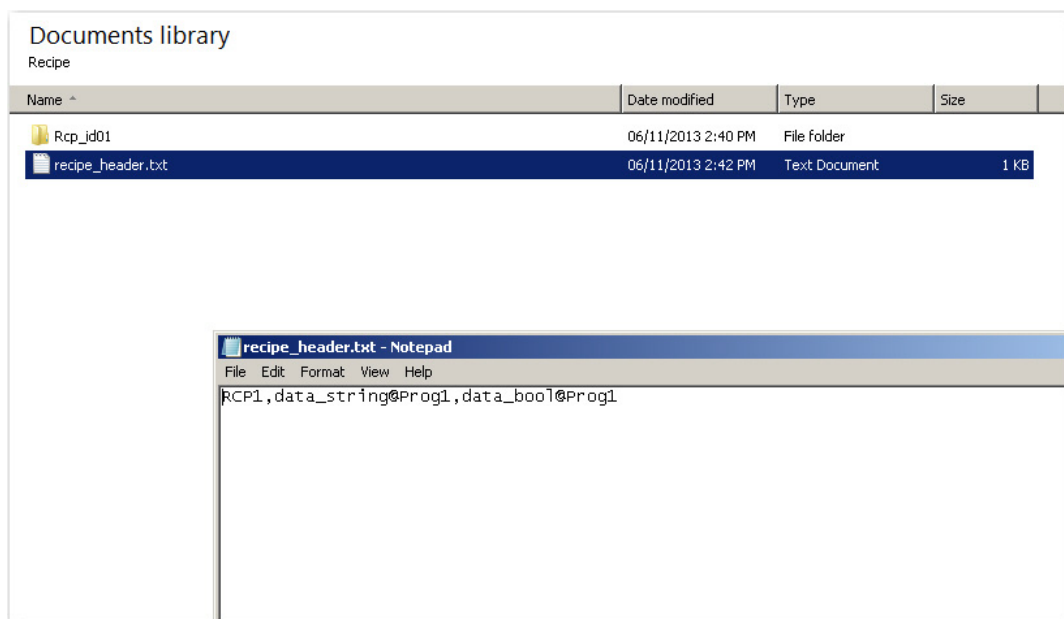
1. Para usar o recurso de Upload, vá para a seção Properties de seu projeto no software Connected Components Workbench.
2. Selecione Recipe. Clique em Manage e escolha Upload. Através do botão Manage, é possível também escolher entre fazer download e excluir os arquivos de receita.
3. A partir da janela Upload que é exibida, selecione o lote de arquivos de receita que deseja carregar.



4. Se o arquivo já existir na sua pasta de destino, escolha entre Overwrite file, Skip file ou Preserve both files.
5. Clique em Upload. A barra de progresso informa se o carregamento foi bem-sucedido ou não.

IMPORTANTE Não remova o cartão microSD do slot enquanto os dados são gravados ou lidos do cartão. As operações de gravação e de leitura em andamento são indicadas pelo LED de status do SD piscando.

Um arquivo de cabeçalho de receita será salvo com as receitas carregadas.



Mapeamento Modbus para Micro800

Mapeamento Modbus

Todos os controladores Micro800 (com exceção dos modelos Micro810 de 12 pontos) suportam Modbus RTU por meio da porta serial incorporada, não isolada. O módulo plug-in de porta serial isolada 2080-SERIALISOL também suporta Modbus RTU. São suportados os Modbus RTU mestre e escravo. Ainda que o desempenho possa ser afetado pelo tempo de varredura do programa, os controladores de 48 pontos podem suportar até seis portas seriais (uma incorporada e cinco plug-in), e, conseqüentemente, seis redes Modbus separadas.

Além disso, os controladores Micro850 e Micro870 suportam o cliente/servidor do Modbus TCP por meio da porta Ethernet.

Configuração Endian

O protocolo Modbus é do tipo big-endian, pois o byte mais significativo de uma palavra de 16 bits é transmitido primeiro. O Micro800 também é do tipo big-endian, portanto, a ordem dos bytes não precisa ser invertida. Para os tipos de dados do Micro800 superiores a 16 bits (por exemplo, DINT, LINT, REAL, LREAL), podem ser necessários múltiplos endereços Modbus, mas o byte mais significativo é sempre o primeiro.

Espaço de endereço de mapeamento e tipos de dados suportados

Como o Micro800 usa nomes de variáveis simbólicas em vez de endereços de memória física, um mapeamento de nome de variável simbólica do para endereçamento Modbus é suportado no software Connected Components Workbench, por exemplo, InputSensorA é mapeado para o endereço Modbus 100001.

Por padrão, o Micro800 segue um endereçamento de seis dígitos especificado na mais recente especificação de Modbus. Por conveniência, conceitualmente o endereço Modbus é mapeado com as seguintes faixas de endereços. A tela de mapeamento do Connected Components Workbench segue esta convenção.

Tabela 51 - Tabela de mapeamento

Tipo de dados variáveis	0 - Bobinas 000001 a 065536		1 - Entradas discretas 100001 a 165536		3 - Registros de entrada 300001 a 365536		4 - Registros de suporte 400001 a 465536	
	Compatível	Endereço Modbus utilizado	Compatível	Endereço Modbus utilizado	Compatível	Endereço Modbus utilizado	Compatível	Endereço Modbus utilizado
BOOL	Y	1	Y	1				
SINT	Y	8	Y	8				
BYTE	Y	8	Y	8				
USINT	Y	8	Y	8				
INT	Y	16	Y	16	Y	1	Y	1
UINT	Y	16	Y	16	Y	1	Y	1
WORD	Y	16	Y	16	Y	1	Y	1
REAL	Y	32	Y	32	Y	2	Y	2
DINT	Y	32	Y	32	Y	2	Y	2
UDINT	Y	32	Y	32	Y	2	Y	2
DWORD	Y	32	Y	32	Y	2	Y	2
LWORD	Y	64	Y	64	Y	4	Y	4
ULINT	Y	64	Y	64	Y	4	Y	4
LINT	Y	64	Y	64	Y	4	Y	4
LREAL	Y	64	Y	64	Y	4	Y	4

NOTA: Strings não são suportadas.

Para tornar mais fácil o mapeamento de variáveis para endereços Modbus de cinco dígitos, a ferramenta de mapeamento do Connected Components Workbench verifica o número de caracteres inseridos para o endereço Modbus. Se apenas cinco dígitos são inseridos, o endereço é tratado como um endereço Modbus de cinco dígitos. Isso significa que as entradas discretas são mapeadas de 00001 a 09999, as bobinas são mapeadas de 10001 a 19999, os registros de entrada são mapeados de 30001 a 39999, e os registros de suporte são mapeados de 40001 a 49999.

Exemplo 1, PanelView 800 IHM (mestre) para Micro800 (escravo)

A porta serial incorporada é destinada para uso com IHMs que usam Modbus RTU. A distância máxima de cabo recomendada é de 3 metros. Use o módulo plug-in de porta serial 2080-SERIALISOL se for necessário ter distâncias maiores ou mais imunidade a ruído.

A IHM é tipicamente configurada para mestre e a porta serial incorporada Micro800 é configurada para escravo.

A partir dos ajustes de parâmetro de comunicações padrão para um PanelView 800 IHM (PV800), existem três itens que devem ser verificados ou modificados para ajustar as comunicações do PV800 para o Micro800.

1. Mude de DF1 para protocolo Modbus.

The screenshot shows the Modbus configuration window. Under the 'Protocol' section, 'Serial' is selected with 'Modbus' in the dropdown. Under 'Driver', 'USB / Ethernet' is selected. The 'Use Ethernet Encapsulation' checkbox is unchecked. The 'PanelView Component Settings' section has a 'Write Optimization' button. Below it is a table with columns 'Port', 'Baud Rate', and 'Data Bits'. The 'Port' is 'RS232', 'Baud Rate' is '19200', and 'Data Bits' is '8'. The 'Controller Settings' section has 'Add Controller' and 'Delete Selected Controller(s)' buttons. A 'Sort by' dropdown is set to 'Name' and 'Ascending'. Below is a table with columns 'Name', 'Controller Type', 'Address', and 'Timing'. The first row is 'PLC-1', 'Modbus', '1', and an empty 'Timing' field.

Port	Baud Rate	Data Bits
RS232	19200	8

Name	Controller Type	Address	Timing
PLC-1	Modbus	1	

2. Defina o endereço do escravo Micro800 para que combine com a configuração da porta serial para o controlador.

The screenshot shows the 'Settings' dialog box. It contains several checkboxes: 'Zero based addressing' (checked), 'Zero based addressing within registers' (checked), 'Holding register bit mask writes' (unchecked), 'Modbus function 06 for single register writes' (checked), 'Modbus function 05 for single coil writes' (checked), 'Default Modbus byte order' (checked), 'First word low in 32 bit data types' (checked), 'First Dword low in 64 bit data types' (checked), and 'Modicon bit ordering (bit 0 is MSB)' (unchecked). There is a 'Close' button at the bottom.

3. Desative tags em erro. Isso serve para evitar a necessidade de ligar e desligar o PV800 quando novos mapeamentos de Modbus forem descarregados do software Connected Components Workbench para o controlador Micro800.

The screenshot shows the 'Modbus TCP Framing' settings. It has two checkboxes: 'Modbus TCP Framing' (unchecked) and 'Deactivate tags on illegal address exception' (unchecked).

Exemplo 2, Micro800 (mestre) para inversor de frequência PowerFlex 4M (escravo)

A seguir estão as características gerais das etapas a ser realizadas para a configuração de um inversor de frequência PowerFlex 4M. Os números de parâmetro listados nesta seção são para um PowerFlex 4M e serão diferentes se você estiver usando um outro inversor de frequência de Classe 4 PowerFlex.

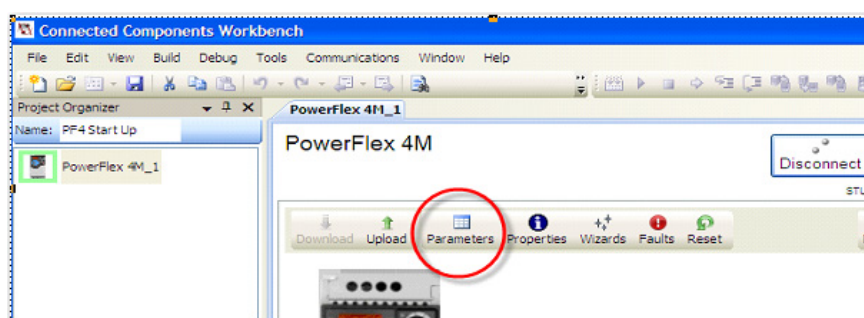
Tabela 52 - Parâmetros

Nome do parâmetro	Número do parâmetro						
	4M	4	40	40P	400	400N	400P
Start Source	P106	P36					
Referência de velocidade	P108	P38					
Comm Data Rate	C302	A103			C103		
Comm Node Addr	C303	A104			C104		
Comm Loss Action	C304	A105			C105		
Comm Loss Time	C305	A106			C106		
Comm Format	C306	A107			C102		

- Conecte o 1203-USB ao inversor PowerFlex e ao computador.
- Inicie o software Connected Components Workbench, conecte ao inversor e defina os parâmetros.

Para configurar o PowerFlex 4M, siga as seguintes etapas:

1. Clique duas vezes no PowerFlex 4M se ainda não estiver aberto no software Connected Components Workbench.
2. Clique em Connect.
3. No navegador de conexão, expanda o driver AB_DF1 DH+™. Selecione AB DSI (porta PF4) e clique em OK.
4. Uma vez que o inversor esteja conectado e tenha sido lido, selecione o assistente de partida e mude os seguintes itens. Selecione Finish para salvar as mudanças para o inversor.
 - Selecione a Porta Comm como a referência de velocidade. Defina P108 [Speed Reference] para 5 (Porta Comm).
 - Defina Start Source para Comm Port. Defina P106 [Start Source] para 5 (Comm Port).
 - Padrões para as entradas remanescentes
 - Aceite os padrões para as restantes e clique em Finish.
5. Selecione Parameters na janela do Connected Components Workbench.



6. A janela de parâmetros se abre. Redimensione-a para visualizar os parâmetros. A partir dessa janela, é possível visualizar os valores de dados definidos dos parâmetros.

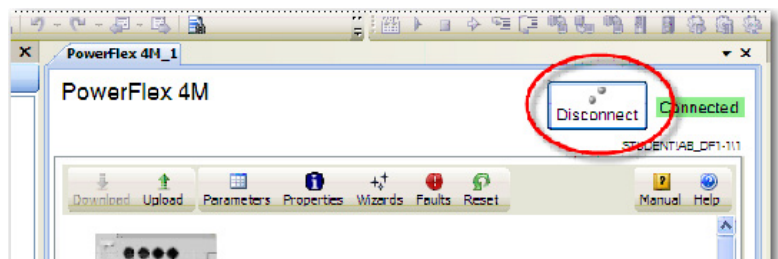
#	Name	Value	Units	Internal Value	Default	Min
1	Output Freq	0.0	Hz	0	0.0	0.0
2	Commanded Freq	0.0	Hz	0	0.0	0.0
3	Output Current	0.00	A	0	0.00	0.00
4	Output Voltage	0.0	V	0	0.0	0.0
5	DC Bus Voltage	314	V	314	0	0
6	Drive Status	0000000000000010		2	00000000000000...	00000000000000...

7. A partir da janela de parâmetros, mude os seguintes parâmetros para definir as comunicações para Modbus RTU, de forma que o inversor PowerFlex 4M se comunique com o Micro830/850/870 via comunicação Modbus RTU.

Tabela 53 - Parâmetros Modbus RTU

Parâmetro	Descrição	Ajuste de parâmetro
C302	Com Data Rate (Baud Rate) 4 = 19.200 bps	4
C303	Communication Node Address (a faixa de endereço é 1 a 127)	2
C304	Com Loss Action (Ação tomada quando há perda de comunicação) 0 = Falha com parada por inércia	0
C305	Com Loss Time (Tempo restante na comunicação antes de tomar a ação definida em C304) 5 s (Máx. 60)	5
C306	Com Format (dados/paridade/parada) RTU:8 Bit de dados, nenhuma paridade, 1 bit de parada	0

8. Desconecte as comunicações e salve seu projeto.



9. Desligue a alimentação para o inversor até que a tela do PowerFlex 4M fique completamente vazia, então recupere a alimentação para o PowerFlex 4M.
Agora o inversor está pronto para ser controlado pelos comandos de comunicação do Modbus RTU iniciados a partir do controlador Micro830/850/870.

Dispositivos Modbus podem ser baseados em 0 (os registros são numerados a partir de 0), ou baseados em 1 (os registros são numerados a partir de 1). Quando inversores de classe 4 PowerFlex são usados com controladores da família Micro800, os endereços de registro listados nos manuais do usuário do PowerFlex precisam ser compensados por n+1.

Por exemplo, a palavra de comando lógico está localizada no endereço 8192, mas seu programa Micro800 precisa usar 8193 (8192+1) para acessá-la.

EXEMPLO: Endereço Modbus (valor n+1 mostrado)

8193	Palavra de comando lógico (parar, iniciar, jog e assim por diante)
8194	Palavra de referência de velocidade
formato xxx.x para 4/4M/40, onde "123" = 12,3 Hz	
formato xxx.xx para 40P/400/400N/400P, onde "123" = 1,23 Hz	
8449	Palavra de status lógico (ler, ativar, falha e assim por diante).
8452	Palavra de realimentação de velocidade (usa o mesmo formato da referência de velocidade)
8450	Palavra de código de erro
(n+1)	Para acessar o parâmetro



Se o respectivo inversor PowerFlex suporta o código de função Modbus 16 pré-selecionado (gravação) múltiplos registros, use uma mensagem de gravação simples de comprimento "2" para gravar o comando lógico (8193) e a referência de velocidade (8194) ao mesmo tempo.

Use um código de função simples 03 ler registros de suporte com um comprimento "4" para ler o status lógico (8449), código de erro (8450), e realimentação de velocidade (8452) ao mesmo tempo.

Consulte o respectivo manual do usuário do inversor PowerFlex de classe 4 para mais informações sobre o endereçamento Modbus. (Consulte o apêndice E - Protocolo Modbus RTU, na publicação [22C-UM001](#)).

Desempenho

O desempenho do MSG_MODBUS (Micro800 é mestre) é afetada pela varredura do programa porque as mensagens são executadas quando a instrução da mensagem é executada em um programa. Por exemplo, se a varredura do programa é 100 ms e são usadas seis portas seriais, então o máximo teórico de portas seriais é de 60 mensagens/segundo total. Esse máximo teórico pode não ser possível porque o MSG_MODBUS é um protocolo de solicitação/resposta mestre/escravo, portanto o desempenho é afetado por diversas variáveis, como o tamanho da mensagem, a baud rate e o tempo de resposta do escravo.

O desempenho do Micro800 quando recebe mensagens de solicitações Modbus (Micro800 é escravo) também é afetado pela varredura do programa. Cada porta serial é executada somente uma vez a cada varredura do programa.

Início rápido

Este capítulo descreve algumas tarefas comuns e instruções de início rápido que têm o objetivo de lhe proporcionar familiaridade com o software Connected Component Workbench. Os seguintes guias rápidos estão incluídos:

Tópico	Página
Faça o upgrade flash do firmware de seu Micro800	265
Estabeleça comunicações entre RSLinx e um controlador Micro830/Micro850/Micro870 por USB	271
Configure a senha do controlador	275
Use o contador em alta velocidade	278
Forçando E/S	289
Usar alteração do modo de operação	291

Faça o upgrade flash do firmware de seu Micro800

Este início rápido lhe mostrará como atualizar o firmware para um controlador Micro800 usando o software Connected Components Workbench, versão 10 ou posterior.

A partir do Software Connected Components Workbench release 10 em diante, há duas opções que você pode selecionar quando atualiza o firmware:

- Fazer Upgrade ou Downgrade – Esta opção mantém a configuração existente do controlador, configurações de Ethernet e senha.
- Reset – Esta opção limpa a configuração existente do controlador, as configurações de Ethernet e a senha.

O procedimento para atualizar o controlador é similar para ambas as opções.



ATENÇÃO: A manutenção da configuração existente do controlador, das configurações de Ethernet e da senha somente está disponível quando se atualiza a revisão do firmware 10 para a mesma revisão ou posterior. Se for atualizar da revisão do firmware 10 para 9 ou anterior, ou atualizar da revisão do firmware 10 a partir de uma revisão anterior, a configuração existente do controlador, as configurações de Ethernet e a senha serão removidas.

IMPORTANTE Se você esqueceu a senha para o controlador, use a opção de Reset para remover a senha.

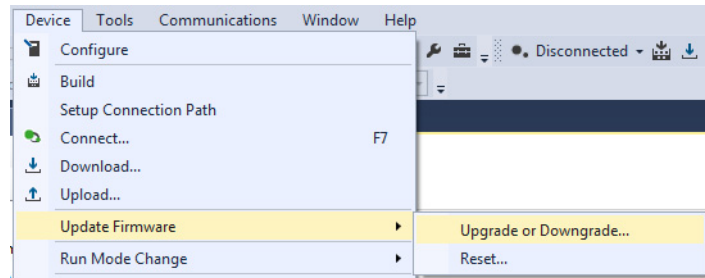
Em controladores Micro850 ou Micro870, os usuários podem usar o upgrade flash em seus controladores por meio da porta ethernet, além da USB.

IMPORTANTE Para atualizar seu controlador com sucesso por USB, conecte somente um controlador ao seu computador, e não realize a atualização em uma máquina virtual como VMware.

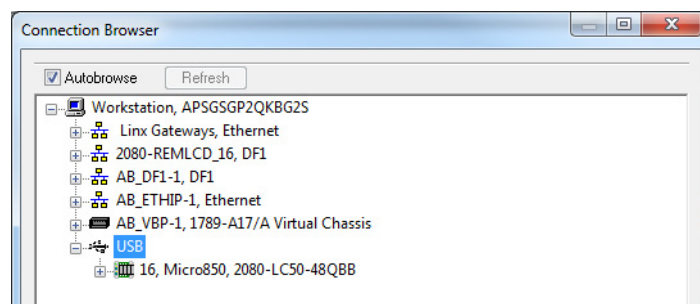
IMPORTANTE Não há suporte para atualização de flash via USB usando o software FactoryTalk Linx com um sistema operacional de 32 bits. Use um sistema operacional de 64 bits ou o software RSLinx Classic.

Para começar, execute o software Connected Components Workbench:

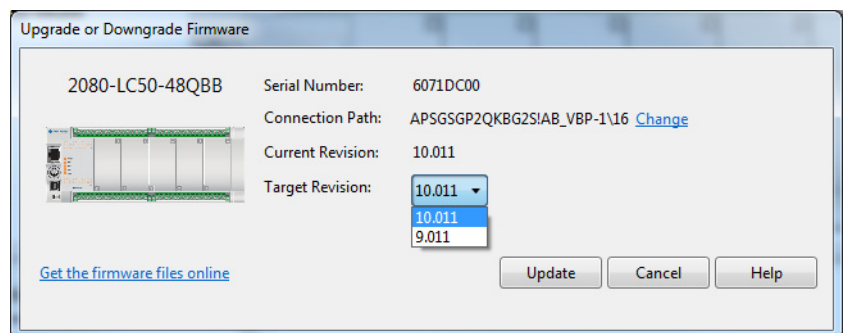
1. No menu, selecione Device → Update Firmware → Upgrade or Downgrade... Ou então, no Project Organizer, clique com o botão direito no controlador e selecione Update Firmware → Upgrade or Downgrade...



2. Se seu projeto não tiver um caminho da conexão para o controlador, a caixa de diálogo do navegador de conexão aparece. Selecione seu controlador, então clique em OK.



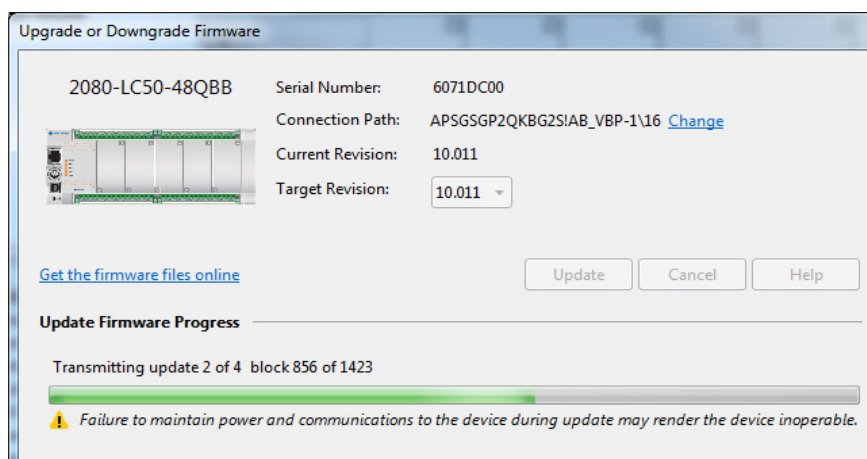
3. Na caixa de diálogo Upgrade or Downgrade Firmware, selecione a revisão-alvo desejada para atualizar o controlador.



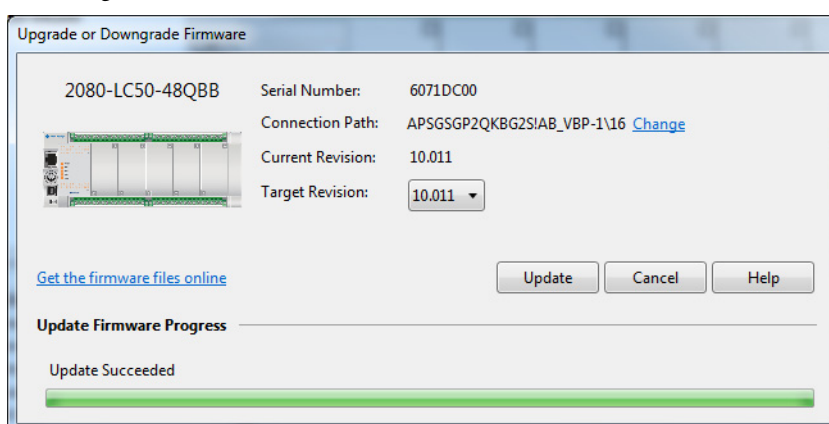
Se a revisão do firmware desejada não for exibida na lista do menu, é possível fazer o download da revisão do firmware clicando no link “Get the firmware files online”.

Você também pode mudar o caminho da conexão clicando no link “Change”.

4. Quando tiver confirmado as configurações, clique em Update para começar a atualização do controlador. O progresso da atualização é mostrado na caixa de diálogo.



5. Depois que a atualização termina, o status é mostrado na caixa de diálogo.



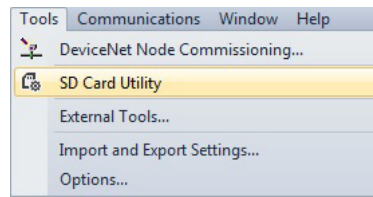
IMPORTANTE Após controlar a atualização do controlador, alguns cartões microSD podem não ser detectados. Remova e insira o cartão microSD ou realize o ciclo de alimentação no controlador se este problema for encontrado.

Atualização flash a partir do cartão MicroSD

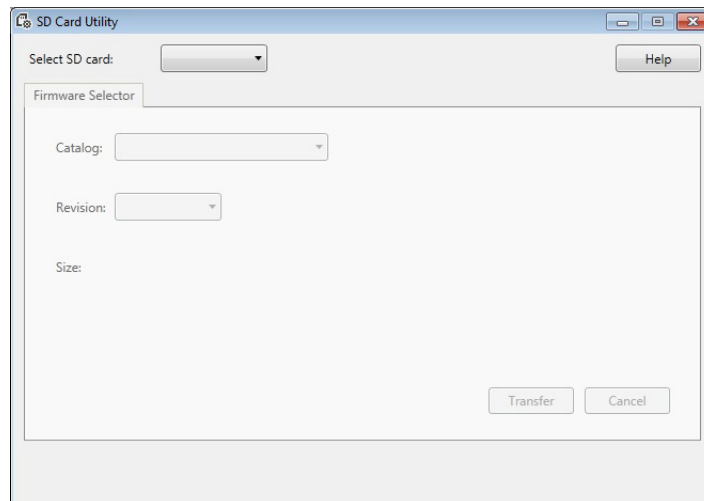
Com o software Connected Components Workbench versão 12 ou posterior e o plug-in de cartão microSD para controladores Micro800, você pode atualizar seu controlador Micro830, Micro850 e Micro870 a partir do cartão microSD, além de usar o ControlFLASH. Este é um processo de duas etapas – primeiro você precisa transferir o firmware para o cartão microSD usando o SD Card Utility, depois precisa editar o arquivo ConfigMeFirst.txt para iniciar o processo de atualização flash. Consulte as instruções a seguir para executar a atualização a partir do cartão microSD.

Etapa 1 – Transferir o firmware para o cartão microSD

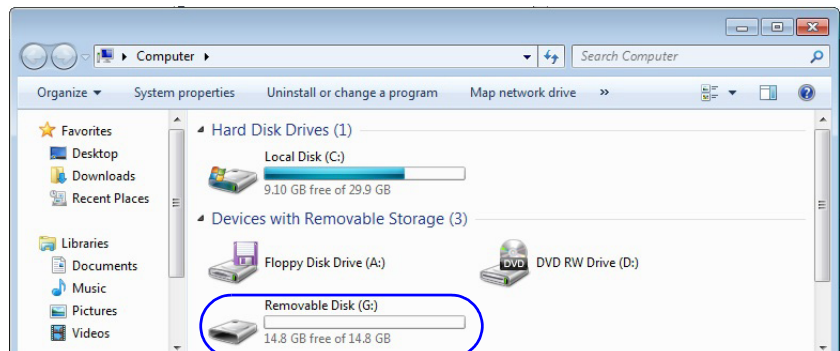
1. Execute o software Connected Components Workbench.
2. Clique em Ferramentas → SD Card Utility.



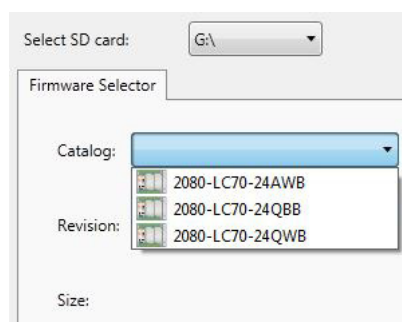
A janela do SD Card Utility é exibida.



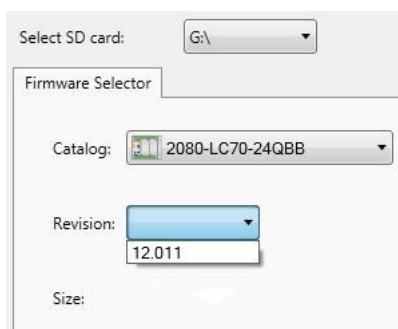
3. Selecione a letra da unidade que aponta para o cartão microSD no seu computador na lista de menu. Você pode verificar a letra da unidade olhando no Windows® Explorer. Para este exemplo, o cartão microSD usa a unidade de letra “G”.



4. Selecione o número do catálogo do seu controlador Micro820.



5. Selecione a versão do firmware com a qual deseja atualizar seu controlador Micro800.

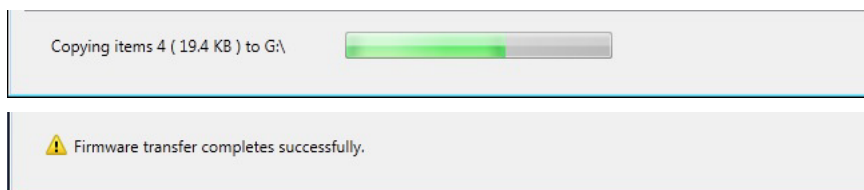


A lista de versões do firmware é instalada juntamente com o software Connected Components Workbench. Se você deseja uma revisão que não está listada, faça o download do firmware do Product Compatibility and Download Center (PCDC) em rok.auto/pcdc e instale o kit ControlFLASH incluído.

IMPORTANTE Você deve fazer login no site da Rockwell Automation antes de fazer o download da versão do firmware.

Feche e inicie novamente o software Connected Components Workbench, em seguida, abra novamente o SD Card Utility. A revisão, agora, deve aparecer na lista.

6. Clique em Transferir.
O arquivo é copiado para o cartão microSD.



7. Feche o SD Card Utility e continue com a próxima etapa para editar o arquivo ConfigMeFirst.txt.

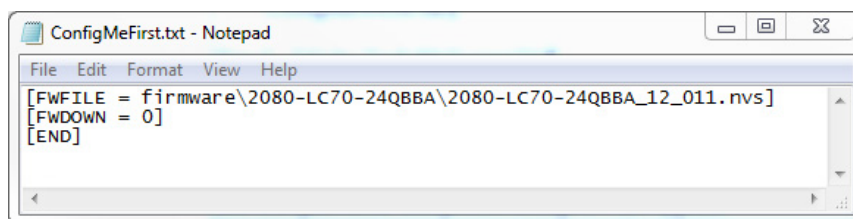
Etapa 2 - Editar o arquivo ConfigMeFirst.txt

Para atualizar o controlador com o firmware que você transferiu para o cartão microSD, é necessário editar o arquivo ConfigMeFirst.txt com os ajustes listados abaixo. Estes ajustes devem ser adicionados no início do arquivo.

Tabela 54 - Novos ajustes de configuração de ConfigMeFirst.txt para atualização flash

Ajuste de parâmetro	Faz efeito em...	Descrição
Configuração da atualização do firmware		
[FWFILE]	Energização	Local do caminho de arquivo da versão do firmware no cartão microSD. O local padrão está no seguinte formato: firmware\<catalog number>\<filename of firmware>
[FWDOWN]	Energização	Define se deve atualizar ou rebaixar o firmware do controlador da versão atual. 0 = Atualizar o firmware; 1 = Rebaixar o firmware IMPORTANTE: A atualização do firmware acontecerá se o ajuste do parâmetro [FWFILE] apontar para uma versão mais nova do arquivo do firmware em comparação com o firmware atual no controlador, independentemente do ajuste do parâmetro [FWDOWN].

Exemplo de arquivo ConfigMeFirst.txt para Flash Upgrade



Depois de ter editado o arquivo, insira o cartão microSD no controlador. Realize o ciclo de alimentação no controlador e o processo de atualização será iniciado. Observe que o LED de status SD não piscará quando a atualização do firmware do cartão microSD estiver em andamento.

Ao usar o software ControlFLASH para fazer o downgrade do firmware de um controlador Micro830 ou Micro850 série B para a versão do firmware 10.011, o programa relata um erro e falha no estágio inicial. No entanto, ao atualizar um controlador Micro800 usando o cartão microSD com uma revisão de firmware que não é compatível com a série, as falhas graves do controlador. Não há nenhum código de erro relatado depois de ter ciclado a alimentação do controlador. O controlador mantém o firmware antigo.

Tabela 55 - Descrição do indicador de status de falha

Estado	Indica
Vermelho sólido	Falha
Verde intermitente	Operação

Para obter uma lista de compatibilidade do firmware e da série, consulte as notas da versão para a versão do firmware 11.011 ou posterior no Product Compatibility and Download Center (PCDC) em rok.auto/pcdc.

Estabeleça comunicações entre RSLinx e um controlador Micro830/Micro850/Micro870 por USB

Este início rápido mostra como fazer o RSLinx® RSWho se comunicar com um controlador Micro830, Micro850 ou Micro870 via USB. Os controladores Micro830, Micro850 e Micro870 usam o driver AB_VBP-x.

RSLinx Classic é instalado como parte do processo de instalação do software Connected Components Workbench. A versão mínima do RSLinx Classic com suporte total do controlador Micro800 é a 2.57, build 15 (lançada em março de 2011).

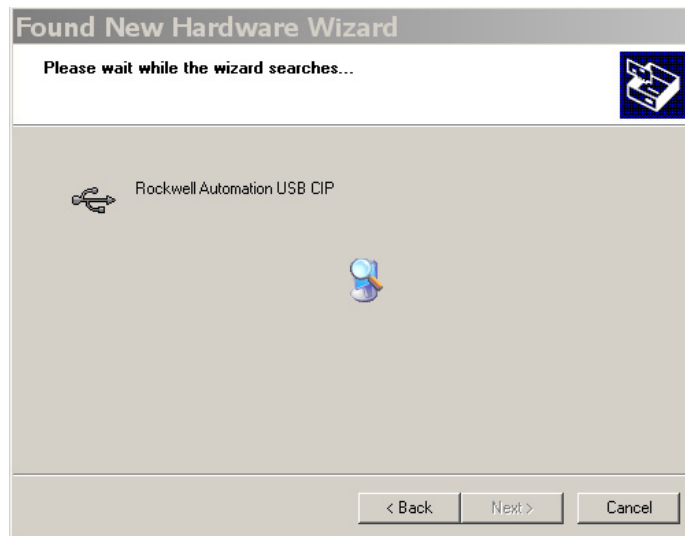
1. Energize o controlador Micro830/Micro850/Micro870.
2. Encaixe o cabo USB A/B diretamente entre seu PC e o controlador Micro830/Micro850/Micro870.
3. O Windows deve detectar o novo hardware. Clique em No, not this time e então clique em Next.



4. Clique em Install the software automatically (recomendado) e, então, clique em Next.



5. O assistente procura o novo hardware.

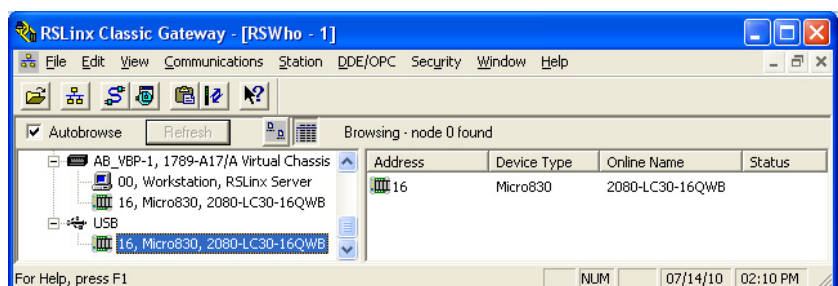


6. Clique em Finish quando o assistente completar a instalação.

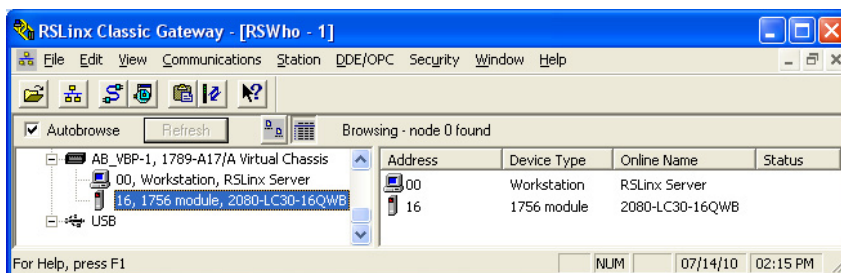


7. Abra o RSLink Classic e execute RSWho clicando no ícone .

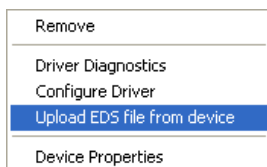
Se o arquivo EDS adequado estiver instalado, o controlador Micro830/ Micro850/Micro870 deve ser identificado de modo apropriado aparecer tanto no driver Virtual Backplane (VBP) e no driver USB, que foi criado automaticamente.



Se, ao contrário, o Micro830/Micro850/Micro870 aparecer como um “Módulo 1756” no driver AB_VBP-1 Virtual Chassis, então o arquivo EDS adequado para esta revisão principal do firmware não foi instalado ou o controlador está executando um pré-lançamento do firmware (revisão principal=0).



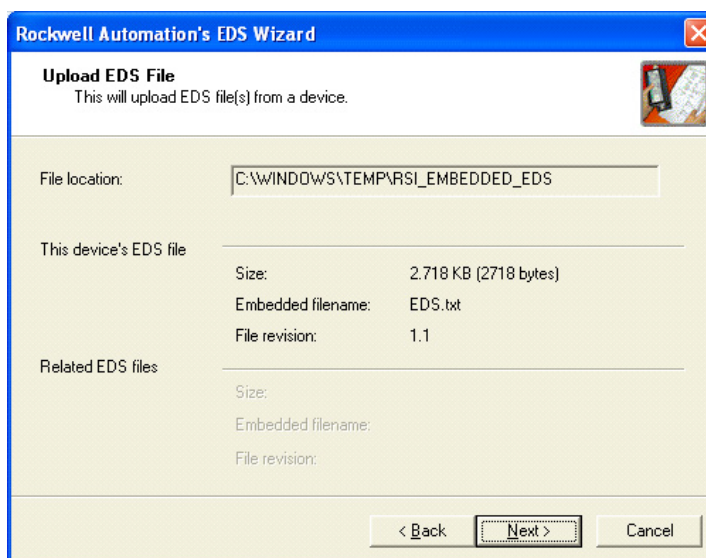
Como os controladores Micro830/Micro850/Micro870 suportam arquivos EDS incorporados, clique com o botão direito nesse dispositivo e selecione Upload EDS file a partir do dispositivo.

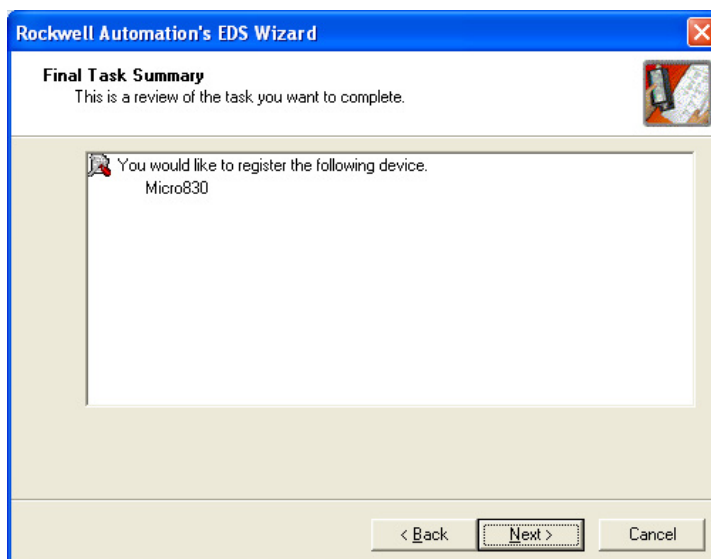
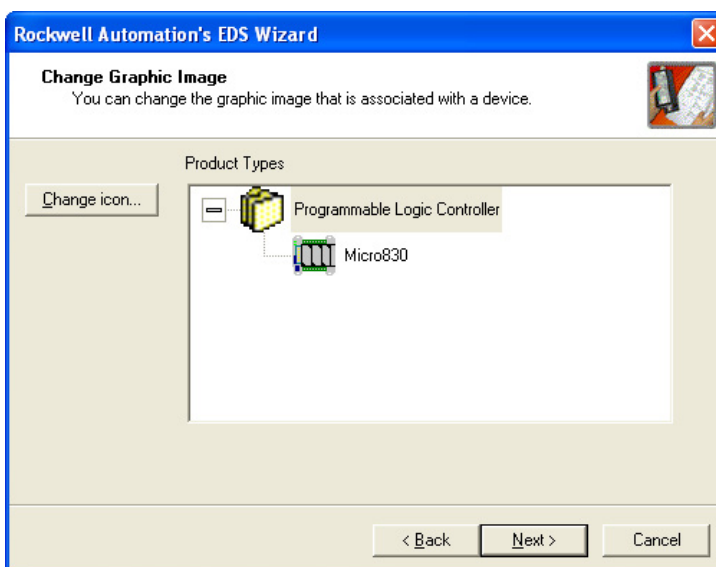
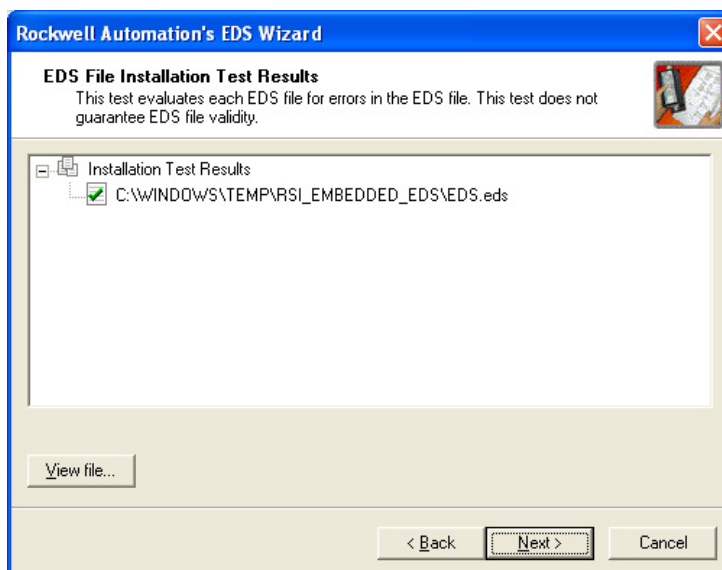


- No assistente EDS que aparecer, clique em Next para continuar.



- Siga os prompts para carregar e instalar o arquivo EDS.

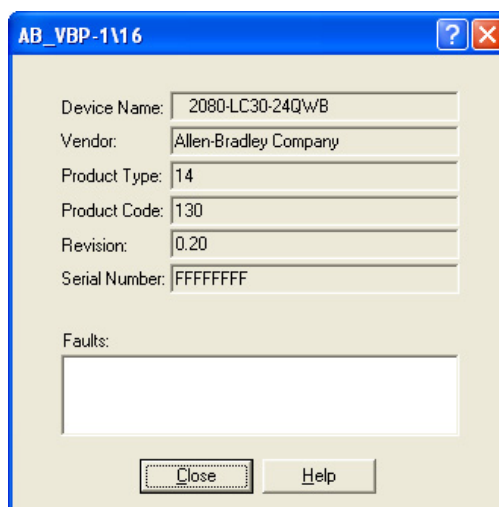




10. Clique em Finish para completar.



Se o Micro830/Micro850/Micro870 ainda aparecer como um módulo 1756, então você provavelmente está executando um pré-lançamento do firmware que está se mostrando como versão principal 0, o que não combina com o arquivo EDS incorporado. Para confirmar, clique com o botão direito no dispositivo e selecione Device Properties (versão do firmware é Major.Minor).



Configure a senha do controlador

Defina, mude e remova a senha em um controlador desejado por meio do software Connected Components Workbench.

IMPORTANTE As instruções a seguir são suportadas na versão 2 do software Connected Components Workbench e controladores Micro800 com versão de firmware 2.

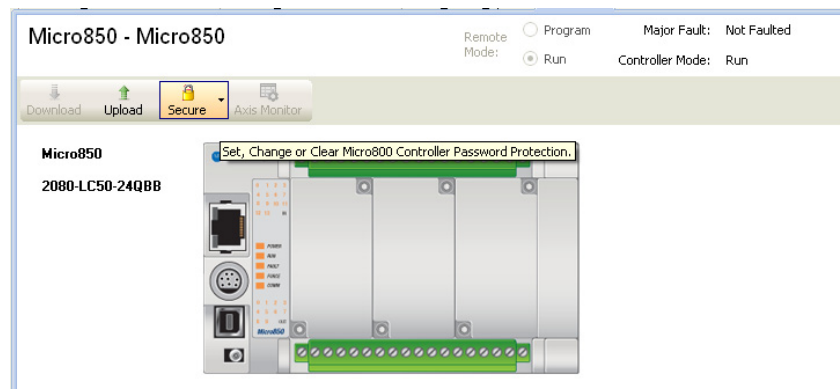
Para mais informações sobre o recurso de senha do controlador em controladores Micro800, consulte [Segurança do controlador na página 221](#).

Defina a senha do controlador

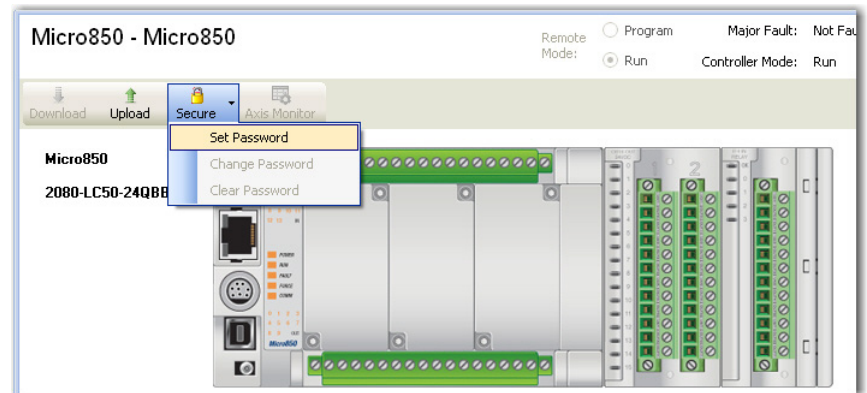
IMPORTANTE Após criar ou modificar a senha do controlador, você precisa desligar o controlador para que a senha seja salva.

Nas instruções a seguir, o software Connected Components Workbench está conectado ao controlador Micro800.

1. No software Connected Components Workbench, abra o projeto para o controlador desejado.
2. Clique em Connect para conectar ao controlador desejado. Na barra de ferramentas Detalhes do dispositivo, é mostrada a mensagem de dica de ferramenta segura “Defina, mude ou remova a proteção por senha do controlador Micro800”.



3. Clique em Secure. Selecione Set Password.



4. Aparece a caixa de diálogo Set Controller Password. Coloque a senha. Confirme a senha fornecendo-a novamente no campo Confirm.



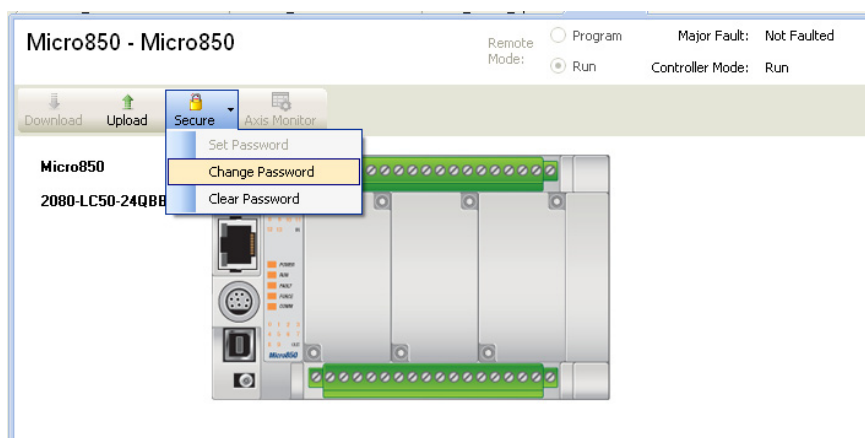
As senhas devem ter ao menos oito caracteres para serem válidas.

5. Clique em OK.
Uma vez que uma senha tenha sido criada, quaisquer novas sessões que tentem conexão com o controlador precisarão fornecer uma senha para ganhar acesso exclusivo ao controlador desejado.

Mude a senha

Com uma sessão autorizada, você pode mudar a senha em um controlador desejado por meio do software Connected Components Workbench. O controlador desejado deve estar no status Connected.

6. Na barra de ferramentas Device Details, clique em Secure. Selecione Change Password.



7. Aparece a caixa de diálogo Change Controller Password. Insira a senha antiga, a senha nova e confirme a nova senha.



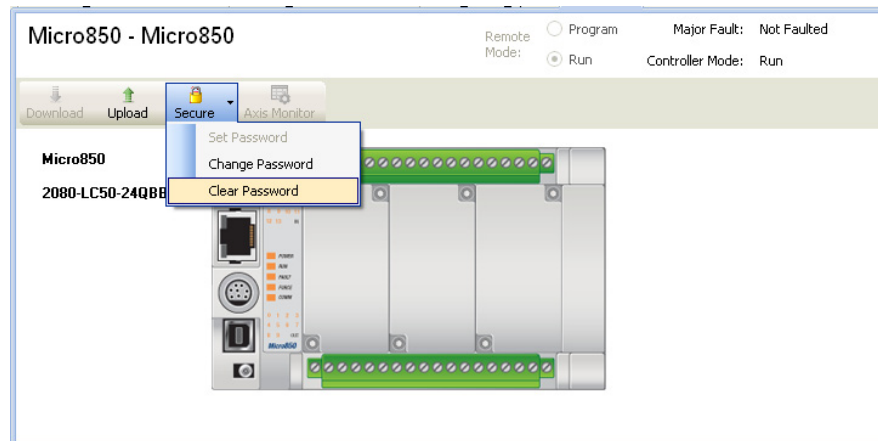
8. Clique em OK.

O controlador exige a nova senha para permitir o acesso a qualquer nova sessão.

Remova a senha

Com uma sessão autorizada, você pode remover a senha em um controlador desejado por meio do software Connected Components Workbench.

1. Na barra de ferramentas Device Details, clique em Secure. Selecione Clear Password.



2. Aparece a caixa de diálogo Clear Password. Insira a senha.
3. Clique em OK para remover a senha.

O controlador não vai exigir nenhuma senha nas próximas novas sessões.

Use o contador em alta velocidade

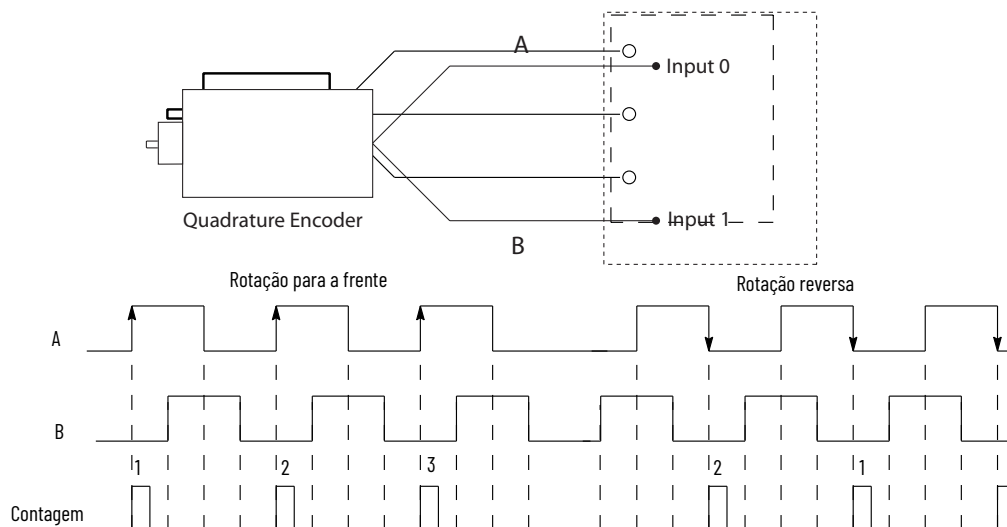
Para usar um dispositivo HSC, primeiro você precisa estabelecer o modo de contagem HSC necessário à sua aplicação. Consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#) para ver os modos disponíveis nos controladores Micro800.

O seguinte exemplo de projeto guia você pela criação de um projeto que usa HSC modo 6, um contador de quadratura com entradas A e B em fase. Ele mostra como escrever um programa ladder simples com o bloco de funções HSC, criar variáveis e atribuir variáveis e valores ao seu bloco de funções. Você também será guiado por um processo de passo a passo sobre como testar seu programa e habilitar um interruptor de luz programável (PLS).

Este exemplo de projeto usa um encoder de quadratura. O encoder de quadratura é utilizado para determinar a direção de rotação e posição para rotação em máquinas como um torno mecânico. O contador bidirecional conta a rotação do encoder de quadratura.

[Figura 25 na página 279](#) mostra um encoder de quadratura conectado às entradas 0 e 1. A direção de contagem é determinada pelo ângulo de fase entre A e B. Se A conduzir B, o contador aumenta. Se B conduzir A, o contador diminui.

Figura 25 - Encoder de quadratura nas entradas 0 e 1

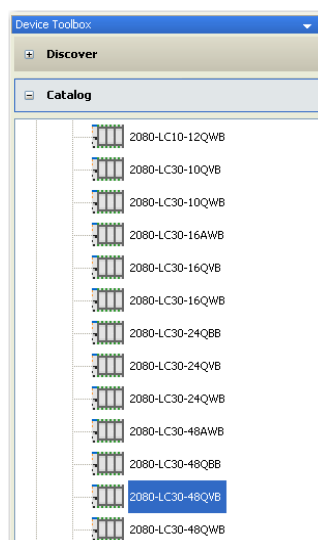


Este início rápido contém com as seguintes seções:

- [Criar o projeto HSC e as variáveis na página 279](#)
- [Atribuir valores às variáveis HSC na página 282](#)
- [Atribuir variáveis ao bloco de funções na página 285](#)
- [Execute o contador em alta velocidade na página 286](#)
- [Use a função de chave de fim de curso programável \(PLS\) na página 288](#)

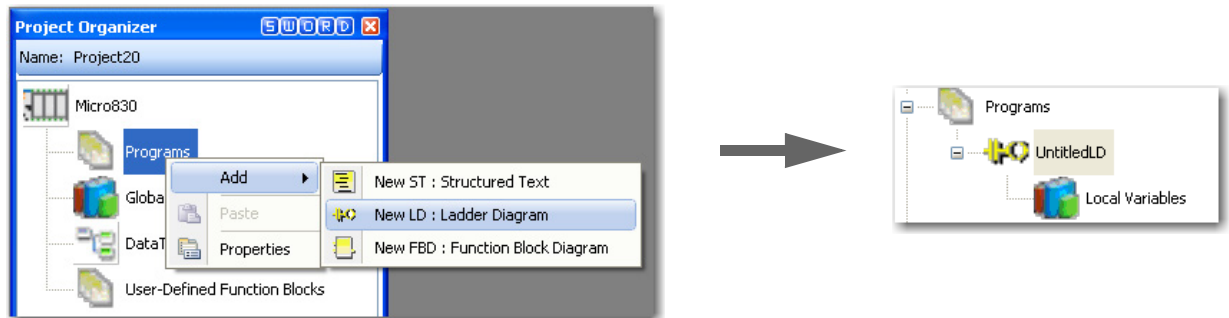
Criar o projeto HSC e as variáveis

1. Inicie o software Connected Components Workbench e abra um novo projeto. A partir da caixa de ferramentas do dispositivo, vá para Catalog → Controllers. Clique duas vezes no seu controlador⁽¹⁾ ou arraste-o e solte-o sobre as janelas do organizador de projeto.

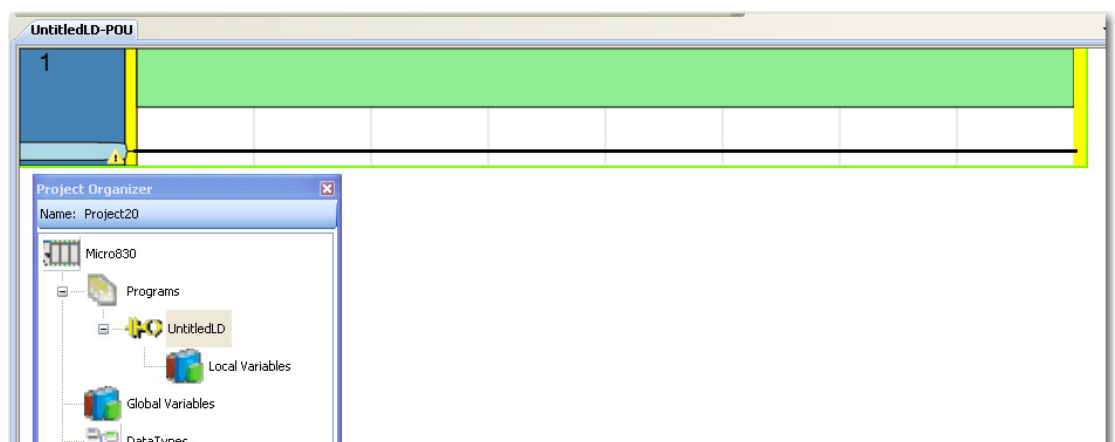


(1) O módulo HSC é suportado em todos os controladores Micro830, Micro850 e Micro870, exceto nos tipos 2080-LCxx-xxAWB.

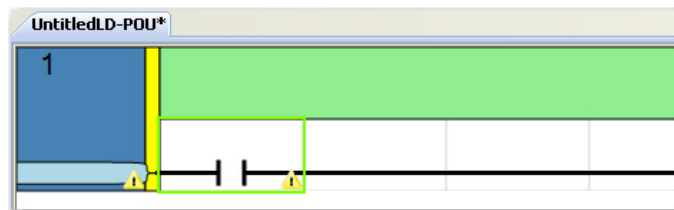
2. Sob o organizador de projetos, clique com o botão direito em Programs. Clique em Add New LD: Ladder Diagram para adicionar um novo programa de lógica ladder.



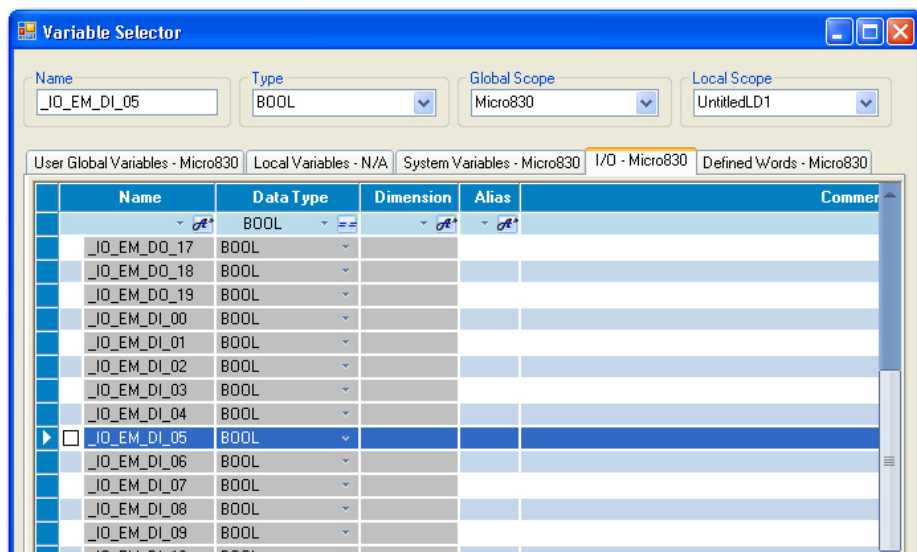
3. Clique com o botão direito em UntitledLD e selecione Open.



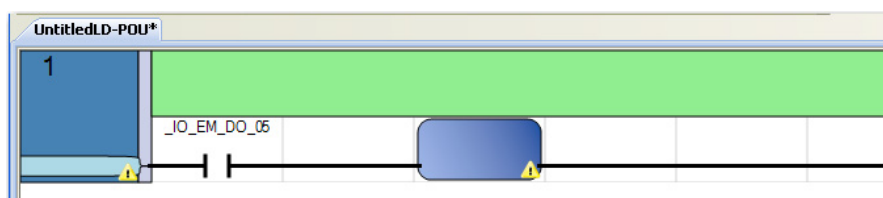
4. Na caixa de ferramentas, clique duas vezes em Direct Contact para adicioná-lo à linha de programa ou arraste e solte o Direct Contact sobre a linha de programa.



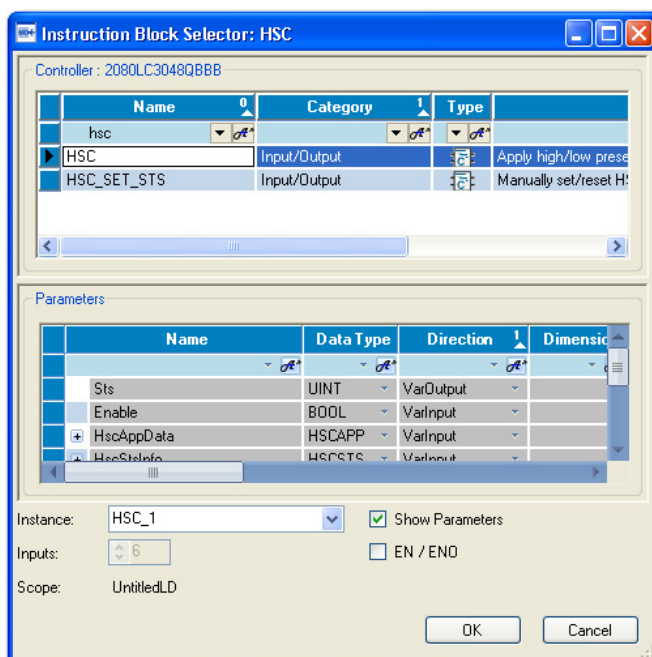
5. Clique duas vezes no Direct Contact que você acabou de adicionar para posicionar a caixa de diálogo Variable Selector. Clique na guia I/O Micro830. Atribua o Direct Contact à entrada 5 selecionando _IO_EM_DI_o5. Clique em OK.



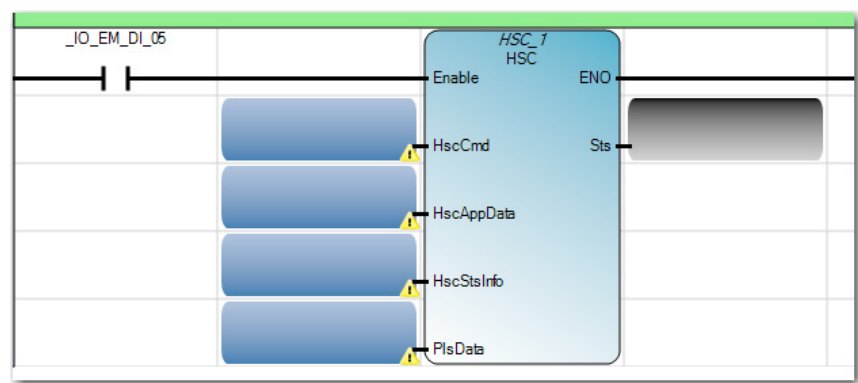
6. À direita do Direct Contact, crie um bloco de funções clicando duas vezes sobre function block na caixa de ferramentas ou arrastando e soltando o bloco de funções sobre a linha.



7. Clique duas vezes no bloco de funções para abrir a caixa de diálogo Instruction Selector. Escolha HSC. Você pode fazer uma rápida pesquisa por um bloco de funções HSC digitando "hsc" no campo Name. Clique em OK.



Sua linha de lógica ladder aparece como mostrado abaixo:

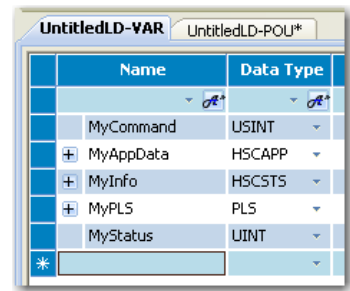


8. No painel Project Organizer, clique duas vezes em Local Variables para posicionar a janela de variáveis. Crie as seguintes variáveis com os tipos de dados correspondentes, como especificado na [Tabela 56 na página 282](#).

Tabela 56 - Tipos de dados de variáveis

Nome da variável	Tipo de dados
MyCommand	USINT
MyAppData	HSCAPP
MyInfo	HSCSTS
MyPLS	PLS
MyStatus	UINT

Após adicionar as variáveis, sua tabela Local Variables deve ter a seguinte aparência:



Atribuir valores às variáveis HSC

Em seguida, você precisa atribuir valores às variáveis que acabou de criar. Tipicamente, uma rotina é usada para atribuir valores às suas variáveis. Para fins de ilustração, este início rápido atribui valores através da coluna Initial Value da tabela Local Variables.

Em um programa real, você deve gravar uma rotina para atribuir valores à sua variável de acordo com sua aplicação.

1. No campo Initial Value para a variável MyCommand, digite 1. Consulte [Comandos HSC \(HScCmd\) na página 212](#) para obter mais informações sobre a descrição de cada valor.
2. Atribua valores às variáveis MyAppData. Expanda a lista das subvariáveis MyAppData clicando no sinal +. Defina os valores das diferentes subvariáveis como mostrado na captura de tela a seguir.

Name	Data Type	Initial Value
HSC_1	HSC	...
MyAppData	HSCAPP	...
MyAppData.PlsEnable	BOOL	FALSE
MyAppData.HscID	UINT	0
MyAppData.HscMode	UINT	6
MyAppData.Accumulator	DINT	
MyAppData.HPSetting	DINT	40
MyAppData.LPSetting	DINT	-40
MyAppData.OFSetting	DINT	50
MyAppData.UFSetting	DINT	-50
MyAppData.OutputMask	UDINT	3
MyAppData.HPOutput	UDINT	1
MyAppData.LPOutput	UDINT	2
MyCommand	USINT	1
MyInfo	HSCSTS	...
MyPLS	PLS	...
MyStatus	UINT	

IMPORTANTE A variável MyAppData tem subvariáveis que determinam os ajustes de parâmetro do contador. É **crucial** conhecer cada uma para determinar como o contador funcionará. Um rápido resumo é fornecido abaixo, mas você também pode consultar [Estrutura de dados HSC APP na página 196](#) para informações detalhadas.

MyAppData.PlsEnable permite ao usuário habilitar ou desabilitar os ajustes de parâmetro PLS. Deve ser definido para FALSE (desabilitado) se a variável MyAppData será usada.

MyAppData.HscID permite ao usuário especificar quais entradas incorporadas serão usadas dependendo do modo e do tipo de aplicação. Consulte a tabela [Entradas HSC e mapeamento da fiação na página 193](#) para conhecer as diferentes IDs que podem ser usadas, assim como as entradas incorporadas e suas características.

Se a ID 0 for usada, a ID 1 não poderá ser usada no mesmo controlador, pois as entradas estão sendo usadas por Reset e Hold.

MyAppData.HscMode permite ao usuário especificar o tipo de operação a qual o HSC usará para contar. Consulte [Modo HSC \(HSCAPP.HSCMode\) na página 197](#) para obter mais informações sobre os modos HSC. Consulte a [Tabela 57](#) para ver a lista de modos disponíveis.

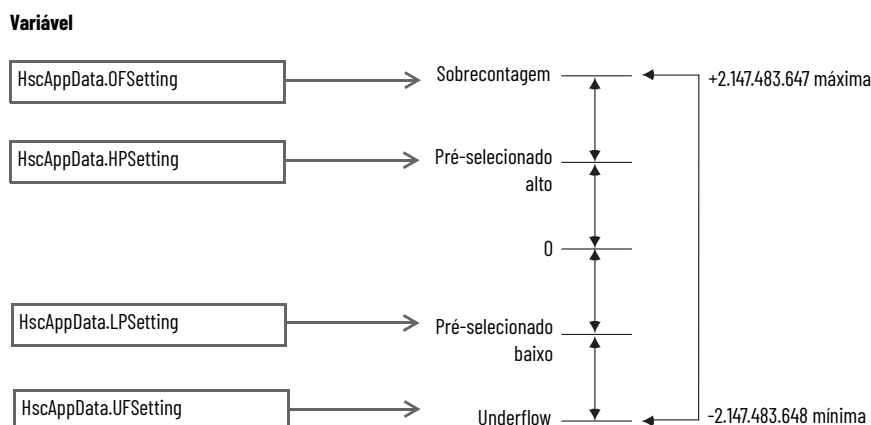
Tabela 57 - Modos de operação HSC

Número do modo	Type [Tipo]
0	Contador crescente - O acumulador é imediatamente desenergizado (0) quando alcança o pré-selecionado alto. Um pré-selecionado baixo não pode ser definido neste modo.
1	Contador crescente com reset e suporte externos - O acumulador é imediatamente desenergizado (0) quando alcança o pré-selecionado alto. Um pré-selecionado baixo não pode ser definido neste modo.
2	Contador com direção externa
3	Contador com direção externa, reset e retenção
4	Contador de duas entradas (para cima e para baixo)
5	Contador de duas entradas (para cima e para baixo) com reset externo e suporte
6	Contador de quadratura (entradas de fase A e B)
7	Contador de quadratura (entradas de fase A e B) com reset externo e suporte
8	Contador X4 de quadratura (entradas de fase A e B)
9	Contador X4 de quadratura (entradas de fase A e B) com reset externo e suporte

Os modos 1, 3, 5, 7 e 9 somente funcionam quando uma ID de 0, 2 ou 4 for definida por causa do fato de que esses modos usam reset e suporte. Os modos 0, 2, 4, 6 e 8 funcionam em qualquer ID. Os modos 6 a 9 somente funcionam quando um encoder está conectado ao controlador. Use a carta de ID HSC como referência para conectar o encoder ao controlador.

MyAppData.HPSetting, **MyAppData.LPSetting**, **MyAppData.OFSetting** e **MyAppData.UFSetting** são todas variáveis definidas pelo usuário que representam a faixa de contagem do HSC. [Figura 26 na página 284](#) traz um exemplo de faixa de valores que pode ser definida para essas variáveis.

Figura 26 - Faixa de valores



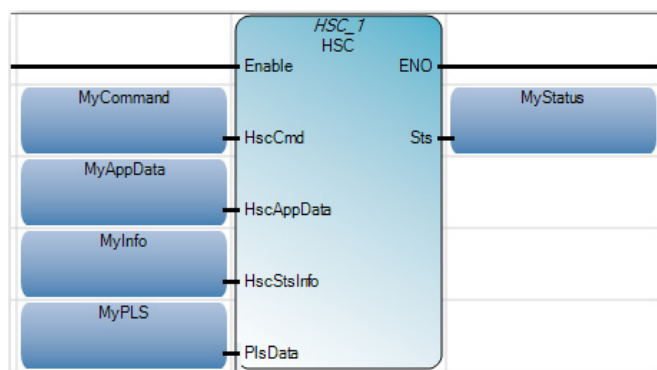
MyAppData.OutputMask, juntamente com **MyAppData.HPOutput** e **MyAppData.LPOutput** permite ao usuário especificar quais saídas incorporadas podem ser ligadas quando um valor pré-selecionado alto ou um valor pré-selecionado baixo é alcançado. Essas variáveis usam uma combinação de números decimais e binários para especificar as saídas incorporadas que são capazes de ligar e desligar.

Assim, no nosso exemplo, primeiro definimos a máscara de saída para um valor decimal de 3 que, quando convertido em binário, é igual a 0011. Isso significa que agora as saídas O0 e O1 podem ser ligadas e desligadas.

Definimos a HPOutput para um valor decimal de 1, que, quando convertido em binário, é igual a 0001. Isso significa que, quando um pré-selecionado alto for alcançado, a saída Oo será ligada e permanecerá ligada até que HSC seja restaurado ou o contador volte a contagem até um pré-selecionado baixo. A LPOutput funciona do mesmo modo que a HPOutput, exceto que uma saída será ligada quando um valor pré-selecionado baixo for alcançado.

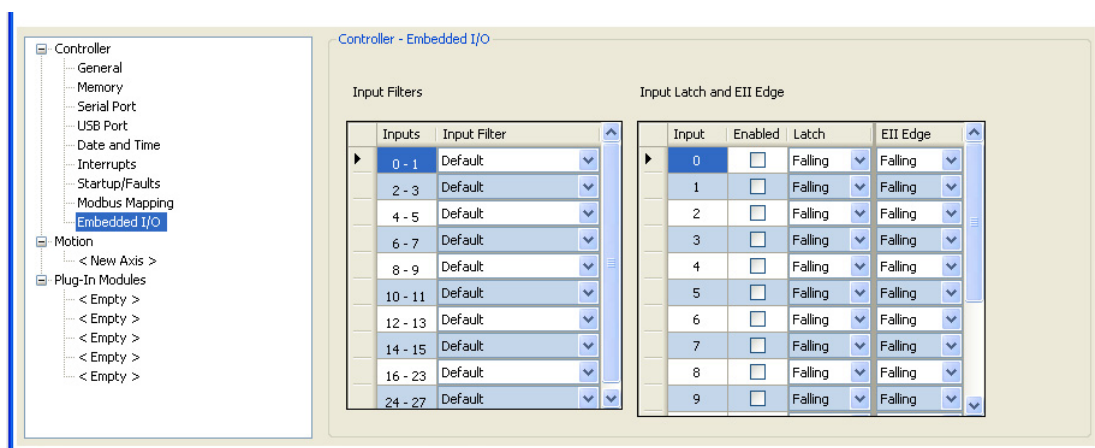
Atribuir variáveis ao bloco de funções

1. Volte ao diagrama de lógica ladder e atribui as variáveis que você acabou de configurar aos elementos correspondentes do bloco de funções HSC. O bloco de funções HSC deve aparecer como mostrado:



Para atribuir uma variável a um elemento em especial no seu bloco de funções, clique duas vezes no bloco vazio da variável. No seletor de variável que aparece, escolha a variável que você acabou de criar. Por exemplo, para o elemento de entrada HSCAppData, selecione a variável MyAppData.

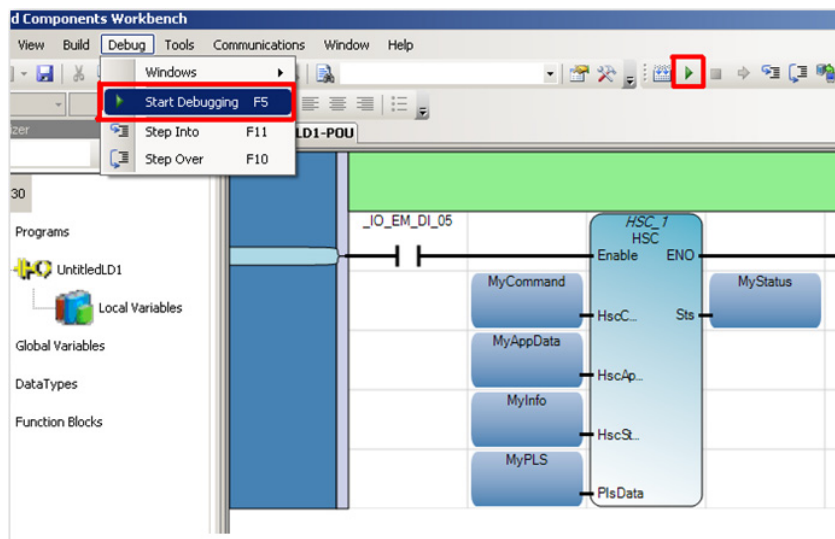
2. Em seguida, clique no controlador Micro830 no painel Project Organizer para posicionar o painel Micro830 Controller Properties. Em propriedades do controlador, clique em Embedded I/O. Defina os filtros de entrada para um valor correto dependendo das características do seu encoder.



3. Certifique-se de que seu encoder está conectado ao controlador Micro830.
4. Energize o controlador Micro830 e conecte-o ao seu computador. Construa o programa no software Connected Components Workbench e faça download dele para o controlador.

Execute o contador em alta velocidade

1. Para testar o programa, vá para o modo de depuração fazendo um dos seguintes:
 - Clique no menu Debug, e então selecione Start Debugging,
 - Clique o botão verde de tocar abaixo da barra do menu, ou
 - Aperte a tecla F5 do Windows.



Agora que estamos no modo de depuração, podemos consultar os valores da saída HSC. O bloco de funções HSC tem duas saídas, uma é a STS (MyStatus) e a outra é a HSCSTS (MyInfo).

2. Clique duas vezes no Direct Contact chamado `_IO_EM_DI_05` para posicionar a janela Variable Monitoring.
3. Clique na guia I/O Micro830. Selecione a sequência `_IO_EM_DI_05`. Selecione as caixas Lock e Logical Value para que esta entrada seja forçada na posição energizada.

The screenshot shows the 'Variable Monitoring' window with the 'I/O - Micro830' tab selected. The table below lists the variables and their status.

Name	Logical Value	Physical Value	Lock	Data Type	Dimension	Alias
_IO_EM_DO_00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_01	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_02	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_03	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_04	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_05	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_06	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_07	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_08	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DO_09	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_00	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_01	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_02	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_03	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_04	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_05	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	BOOL		
_IO_EM_DI_06	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_07	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_08	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_09	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_12	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		
_IO_EM_DI_13	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	BOOL		

4. Clique na guia Local Variables para ver quaisquer mudanças em tempo real sendo feitas às variáveis. Expanda a lista de variáveis MyAppData e MyInfo clicando no sinal +.
5. Ligue o encoder para ver o contador contar para cima/para baixo. Por exemplo, se o encoder está conectado a um eixo do motor, então ligue o motor para disparar a contagem HSC. O valor do contador será mostrado no MyInfo.Accumulator. A variável MyStatus deve mostrar um valor lógico de 1, o que significa que o HSC está funcionando.



Consulte [Códigos de status de bloco de funções HSC na página 213](#) para a lista completa de códigos de status. Por exemplo, se o valor de MyStatus for 4, há um erro de configuração e o controlador irá. Nesse caso, você precisa verificar os parâmetros.

Variable Monitoring			
Global Variables - Micro830 Local Variables - UntitledLD1 System Variables - Micro830 I/O - Micro830			
Name	Logical Value	Physical Value	Initial Value
HSC_1
MyCommand	1	N/A	1
MyAppData
MyAppData.PlsEnable	<input type="checkbox"/>	N/A	FALSE
MyAppData.HscID	0	N/A	0
MyAppData.HscMode	7	N/A	5
MyAppData.Accumulator	40	N/A	
MyAppData.HPSSetting	40	N/A	40
MyAppData.LPSSetting	-40	N/A	-40
MyAppData.OFSSetting	50	N/A	50
MyAppData.UFSSetting	-50	N/A	-50
MyAppData.OutputMask	3	N/A	3
MyAppData.HPOutput	1	N/A	1
MyAppData.LPOutput	2	N/A	2
MyInfo
MyInfo.CountEnable	<input checked="" type="checkbox"/>	N/A	
MyInfo.ErrorDetected	<input type="checkbox"/>	N/A	
MyInfo.CountUpFlag	<input checked="" type="checkbox"/>	N/A	
MyInfo.CountDwnFlag	<input checked="" type="checkbox"/>	N/A	
MyInfo.ModeTDone	<input type="checkbox"/>	N/A	
MyInfo.OVF	<input type="checkbox"/>	N/A	
MyInfo.UNF	<input type="checkbox"/>	N/A	
MyInfo.CountDir	<input checked="" type="checkbox"/>	N/A	
MyInfo.HPReached	<input checked="" type="checkbox"/>	N/A	
MyInfo.LPReached	<input type="checkbox"/>	N/A	
MyInfo.OFCauseInter	<input type="checkbox"/>	N/A	
MyInfo.UFCauseInter	<input type="checkbox"/>	N/A	
MyInfo.HPCauseInter	<input type="checkbox"/>	N/A	
MyInfo.LPCauseInter	<input type="checkbox"/>	N/A	
MyInfo.PlsPosition	0	N/A	
MyInfo.ErrorCode	0	N/A	
MyInfo.Accumulator	40	N/A	
MyInfo.HP	40	N/A	
MyInfo.LP	-40	N/A	
MyInfo.HPOutput	1	N/A	
MyInfo.LPOutput	2	N/A	
MyPLS
MyStatus	1	N/A	

Para este exemplo, uma vez que o acumulador alcança o valor pré-selecionado alto de 40, a saída o liga e a flag HPReached liga. Uma vez que o acumulador alcança o valor pré-selecionado baixo de -40, a saída 1 liga e o flag LPReached também liga.

Use a função de chave de fim de curso programável (PLS)

A função chave de fim de curso programável permite configurar o contador de alta velocidade para operar como uma PLS (chave de fim de curso programável) ou uma chave de came rotativa. A PLS é utilizada quando você precisa de mais de um par de pré-selecionados altos e baixos (até 255 pares de pré-selecionados altos e baixos são suportados pela PLS).

1. Inicie um novo projeto seguindo as mesmas etapas e valores do projeto anterior. Defina os valores para as seguintes variáveis como segue:
 - A variável HSCAPP.PlsEnable deve ser definida como TRUE.
 - Defina um valor somente para UFSetting e OFSetting (a OutputMask é opcional dependendo se uma saída deve ou não ser definida). Seus novos valores devem seguir o exemplo da [Figura 27](#):

Figura 27 - Valores PLS

UntitledLD1-VAR						
	Name	Data Type	Dimension	Alias	Initial Value	Attribute
+	HSC_1	HSC			...	ReadWrite
	MyCommand	USINT			1	ReadWrite
-	MyAppData	HSCAPP			...	ReadWrite
	MyAppData.PlsEnable	BOOL			TRUE	ReadWrite
	MyAppData.HscID	UINT			0	ReadWrite
	MyAppData.HscMode	UINT			7	ReadWrite
	MyAppData.Accumulator	DINT				ReadWrite
	MyAppData.HPSetting	DINT				ReadWrite
	MyAppData.LPSetting	DINT				ReadWrite
	MyAppData.OFSetting	DINT			50	ReadWrite
	MyAppData.UFSetting	DINT			-50	ReadWrite
	MyAppData.OutputMask	UDINT			255	ReadWrite
	MyAppData.HPOutput	UDINT				ReadWrite
	MyAppData.LPOutput	UDINT				ReadWrite
+	MyInfo	HSCSTS			...	ReadWrite
-	MyPLS	PLS	[1..4]		...	ReadWrite
	MyPLS[1]	PLS			...	ReadWrite
	MyPLS[1].HscHP	DINT			10	ReadWrite
	MyPLS[1].HscLP	DINT			-10	ReadWrite
	MyPLS[1].HscHPOutPut	UDINT			1	ReadWrite
	MyPLS[1].HscLPOutPut	UDINT			16	ReadWrite
	MyPLS[2]	PLS			...	ReadWrite
	MyPLS[2].HscHP	DINT			20	ReadWrite
	MyPLS[2].HscLP	DINT			-20	ReadWrite
	MyPLS[2].HscHPOutPut	UDINT			2	ReadWrite
	MyPLS[2].HscLPOutPut	UDINT			32	ReadWrite
	MyPLS[3]	PLS			...	ReadWrite
	MyPLS[3].HscHP	DINT			30	ReadWrite
	MyPLS[3].HscLP	DINT			-30	ReadWrite
	MyPLS[3].HscHPOutPut	UDINT			4	ReadWrite
	MyPLS[3].HscLPOutPut	UDINT			64	ReadWrite
	MyPLS[4]	PLS			...	ReadWrite
	MyPLS[4].HscHP	DINT			40	ReadWrite
	MyPLS[4].HscLP	DINT			-40	ReadWrite
	MyPLS[4].HscHPOutPut	UDINT			8	ReadWrite
	MyPLS[4].HscLPOutPut	UDINT			128	ReadWrite
	MyStatus	UINT				ReadWrite

Neste exemplo, a variável PLS recebe uma dimensão de [1..4]. Isso significa que o HSC pode ter quatro pares de pré-selecionados altos e baixos.

Mais uma vez, seus pré-selecionados altos devem ser definidos abaixo do OFSetting e o pré-selecionado baixo deve ser mais alto que o UFSetting. Os valores HscHPOutPut e HscLPOutPut determinarão quais saídas serão ligadas quando um pré-selecionado alto ou baixo for alcançado.

2. Agora você pode construir e descarregar o programa no controlador e então depurá-lo e testá-lo seguindo as instruções para o último projeto.

Forçando E/S

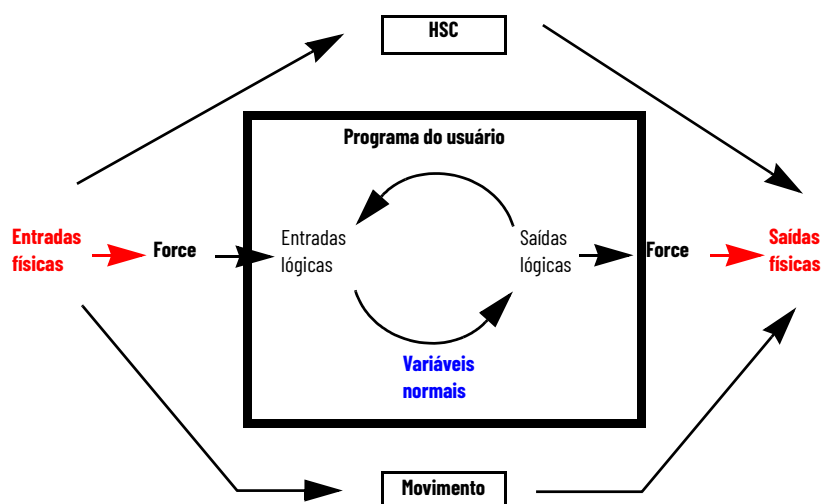
As entradas são forçadas logicamente. Os indicadores de status não mostram valores forçados, mas as entradas no programa do usuário são forçadas.

Forçar somente é possível com E/S e não se aplica a variáveis definidas pelo usuário e variáveis não E/S e funções especiais como HSC e movimento, que são executadas independentemente da varredura do programa do usuário. Por exemplo, para movimento, a entrada de inversor pronto não pode ser forçada.

Diferentemente das entradas, as saídas são forçadas fisicamente. Os indicadores de status mostram valores forçados e o programa do usuário não usa valores forçados.

[Figura 28](#) ilustra o comportamento forçado.

Figura 28 - Forçar comportamento de E/S



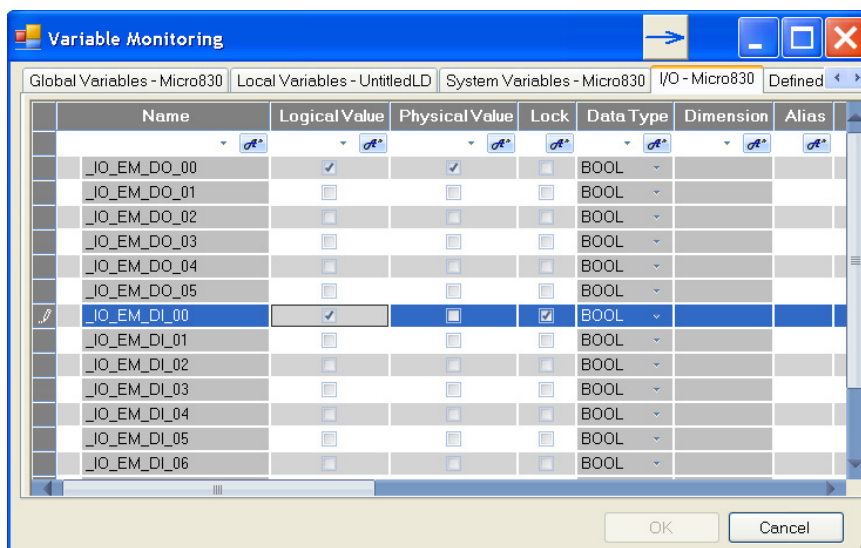
- Os indicadores de status LED sempre combinam com o valor físico da E/S
- Variáveis internas não físicas normais não podem ser forçadas
- Funções especiais como HSC e movimento não podem ser forçadas



ATENÇÃO: Forçar variável pode resultar em um movimento repentino da máquina, possivelmente ferindo pessoal ou danificando equipamentos. Use muito cuidado quando forçar variáveis.

Verificar se as forças (travas) estão habilitadas

Se o software Connected Components Workbench estiver disponível, verifique o monitor de variáveis enquanto estiver depurando online. O forçamento é realizado primeiro travando uma variável E/S e então definindo o valor lógico para entradas e o valor físico para saídas. Lembre-se de que não é possível forçar uma entrada física nem uma saída lógica.



Em muitos casos, a frente do controlador não é visível ao operador e o software Connected Components Workbench não está online com o controlador. Se você deseja que o status forçado seja visível ao operador, então o programa do usuário deve mostrar o status forçado usando o bloco de funções SYS_INFO e então mostrar o status forçado em algo que o operador pode visualizar, como a interface de operação e programação (IHM), ou a torre luminosa. [Figura 29](#) é um exemplo de programa em Texto Estruturado.

Figura 29 - Exemplo de programa em texto estruturado

```

1  (* Read System Information including Force Enable bit *)
2  SYS_INFO_1(TRUE);
3
4  (* Turn on Warning Light if Forces are Enabled *)
5  If SYS_INFO_1.Sts.ForcesInstall = TRUE THEN
6    _IO_EM_DO_05 := TRUE;
7  ELSE
8    _IO_EM_DO_05 := FALSE;
9  END_IF;

```

Se a frente do controlador está visível, e não bloqueada pelo gabinete, os controladores Micro830, Micro850 e Micro870 têm um indicador LED de força.

Force de E/S após um ciclo de energização/desenergização

Depois que um controlador é desligado e ligado, todas as forças de E/S são removidas da memória.

Usar alteração do modo de operação

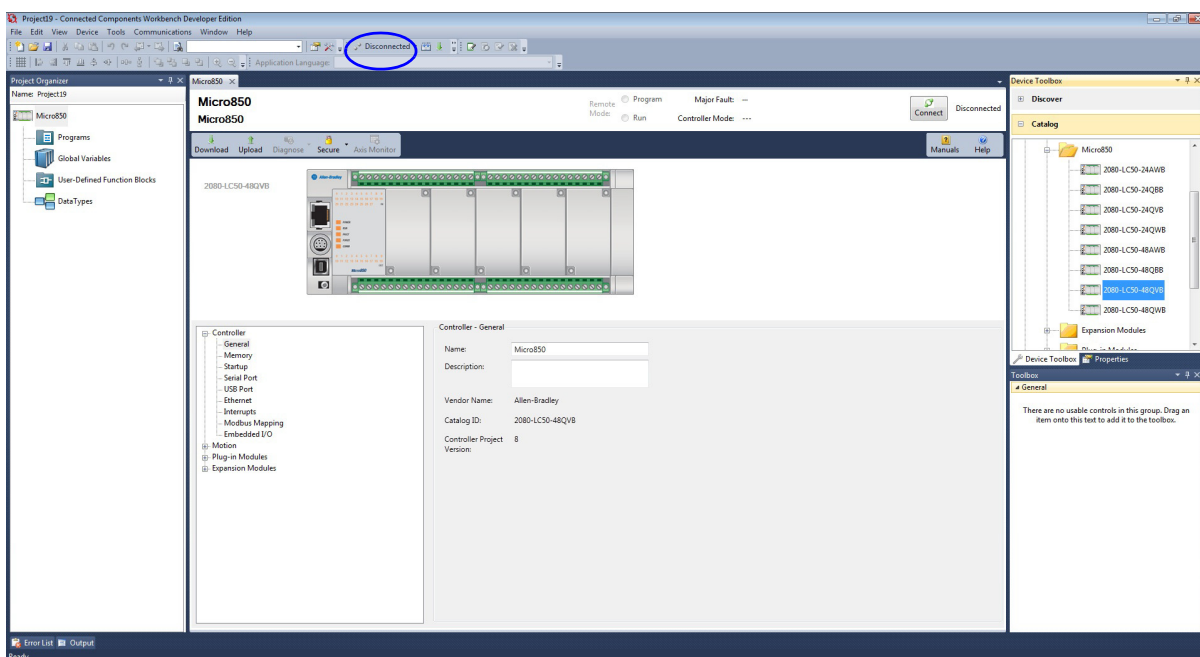
A alteração do modo de operação permite ao usuário fazer pequenas alterações na lógica de um projeto em funcionamento e testá-las imediatamente no controlador, sem precisar entrar no modo de programa nem se desconectar do controlador.

- IMPORTANTE** As seguintes exigências devem ser atendidas para usar a alteração de modo de operação:
- Versão do firmware do controlador Micro820/Micro830/Micro850 8.0 ou posterior, e
 - Software Connected Components Workbench Developer Edition, versão 8.0 ou posterior.

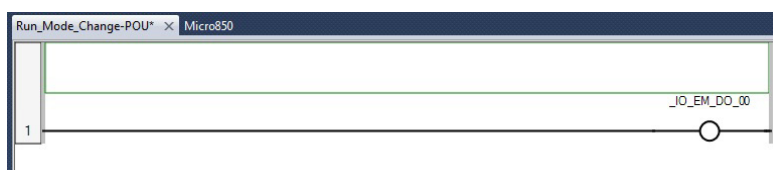
O seguinte exemplo de projeto guia você pela criação de uma aplicação simples para um controlador Micro850 sem nenhum módulo plug-in, e mostra como usar o recurso de alteração do modo de operação.

Criação do projeto

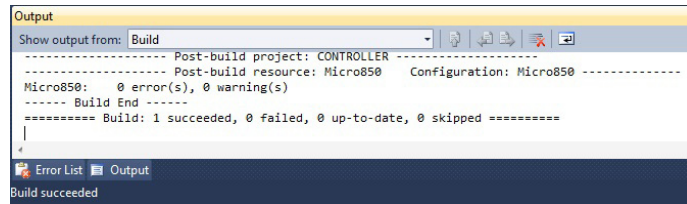
1. Crie um novo projeto para um controlador Micro830/Micro850/Micro870 sem nenhum plug-in. Observe que o controlador está desconectado.



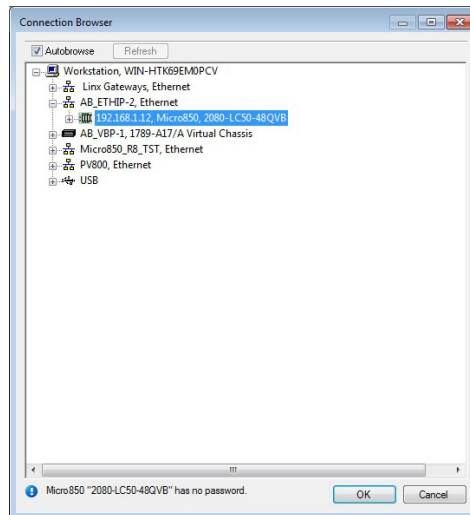
2. Clique com o botão direito em Programas e selecione Add → New LD: Ladder Diagram.
3. Na caixa de ferramentas, clique duas vezes em Direct Coil para adicioná-lo à linha ou arraste e solte Direct Coil sobre a linha.
4. Clique com o botão direito na recentemente adicionada Bobina direta para fazer aparecer a caixa de diálogo Variable Selector e selecione “_IO_EM_DO_00”.



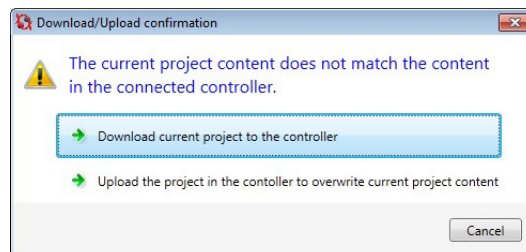
5. Construa o projeto.



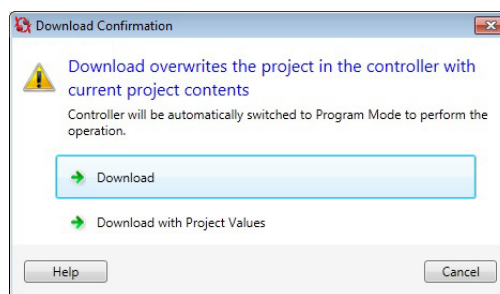
6. Faça o download do projeto para o controlador.
Na caixa de diálogo do Connection Browser, selecione o controlador Micro850.



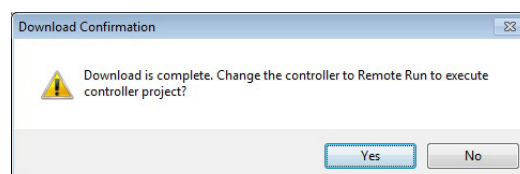
7. Selecione Download current project to the controller.



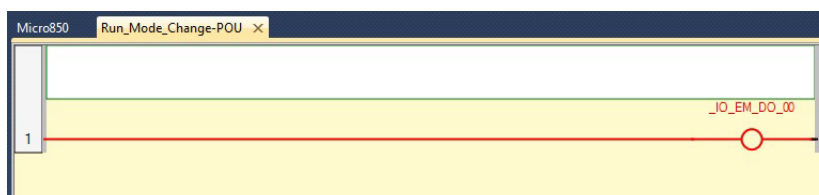
8. Selecione Download to confirm.



9. Quando o projeto tiver sido descarregado para o controlador, aparecerá uma caixa de diálogo pedindo para alterar o controlador para funcionamento remoto. Clique em Yes.



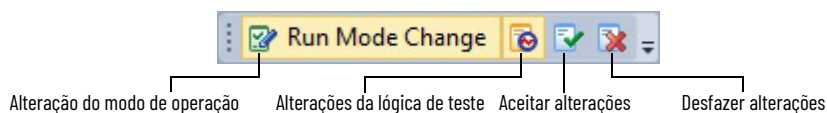
10. Observe que o controlador está, agora, no modo Debug.




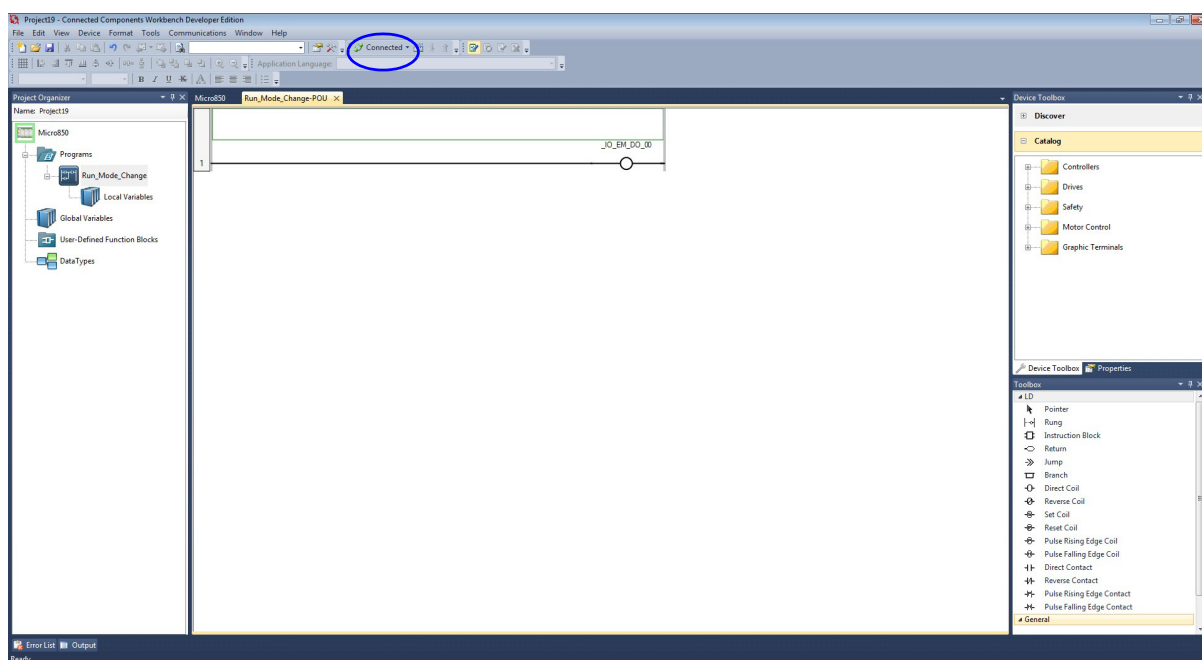
IMPORTANTE A partir do software Connected Components Workbench versão 8.0 em diante, selecionar “Yes” para mudar o controlador para o modo de operação remoto após o download de um projeto, muda-o automaticamente para o modo de depuração.

Edite o projeto usando Alterar o modo de operação

Barra de ferramentas de alteração do modo de operação



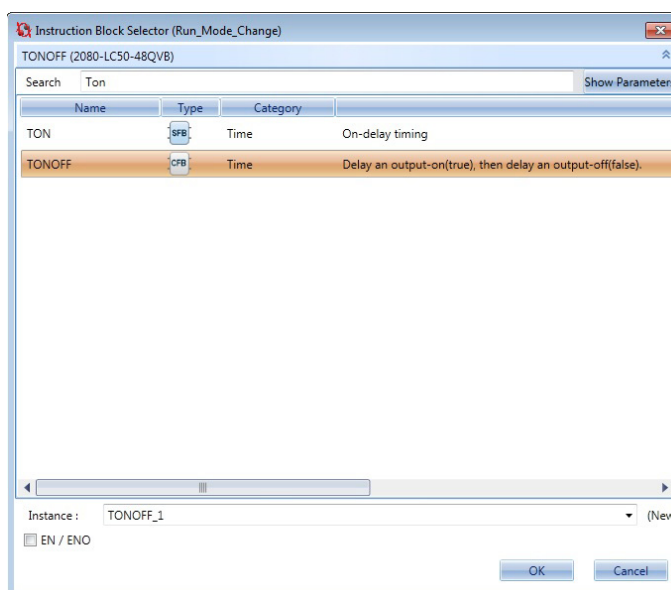
1. Clique no ícone  de alteração do modo de operação. Observe se o controlador entra no modo Edit e ainda está conectado.



Se você adicionar uma nova variável durante RMC, não estarão disponíveis o acesso aos dados externos e a modificação do tipo de acesso (o padrão é Ler/Escrever) dessa nova variável até que você tenha escolhido aceitar ou desfazer as alterações da lógica de teste.

2. Na caixa de ferramentas, clique duas vezes em Instruction Block para adicioná-lo à linha ou arraste e solte Instruction Block sobre a linha.

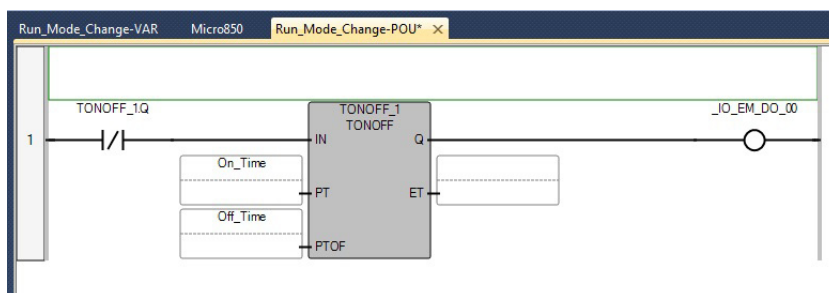
3. Clique duas vezes no bloco de instruções recentemente adicionado e selecione “Timer On/Off” (TONOFF).




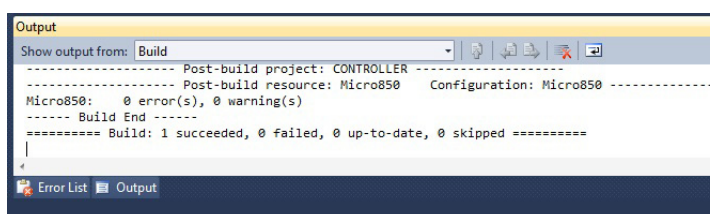
Configure o bloco de instruções para disparar a cada segundo.

Name	Alias	Data Type	Dimension	Project Value	Initial Value	Comment	String Size
TONOFF_1		TONOFF			
On_Time		TIME			T#1s		
Off_Time		TIME			T#1s		

4. Na caixa de ferramentas, clique duas vezes em Reverse Contact para adicioná-lo à linha ou arraste e solte Reverse Contact sobre a linha. Coloque-o à esquerda do bloco de instruções recentemente adicionado.

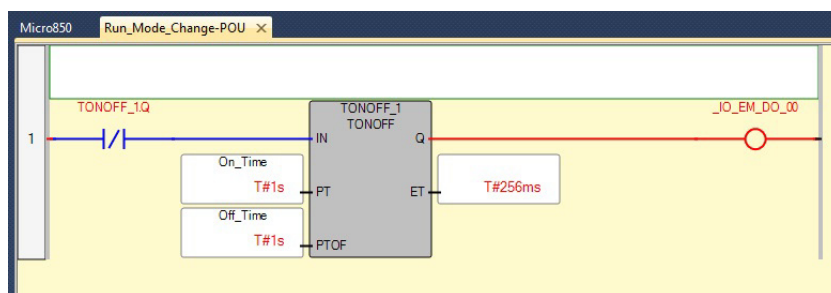


5. Clique no ícone Test Logic Changes  para construir o projeto e descarregá-lo para o controlador.




IMPORTANTE Quando é realizado um Testar lógica, ou ao desfazer as alterações depois que se completa um Testar lógica, quaisquer instruções de comunicação ativas serão abortadas enquanto as alterações são descarregadas para o controlador.

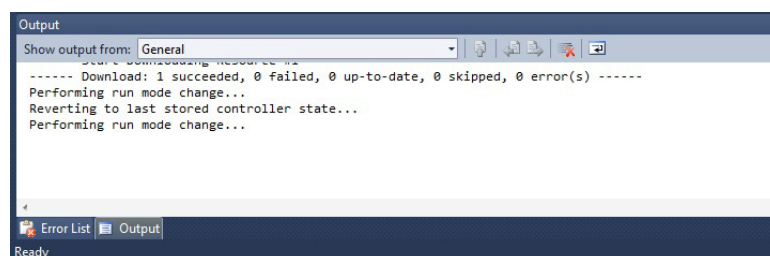
- O controlador entrará automaticamente no modo de depuração e exibirá o projeto atualizado.



- Agora você pode escolher desfazer ou aceitar as alterações do projeto.

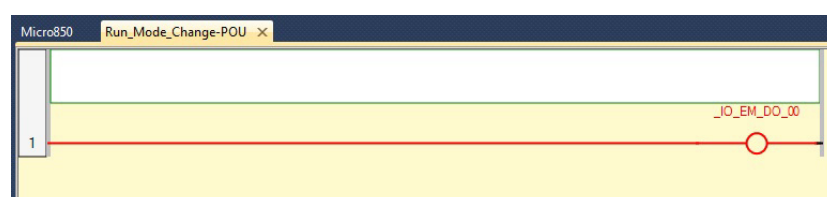
Para desfazer as alterações

- Clique no ícone  Undo Changes.
- As alterações serão descartadas e o projeto original será restaurado no controlador.




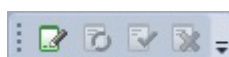
IMPORTANTE Quando é realizado um Testar lógica, ou ao desfazer as alterações depois que se completa um Testar lógica, quaisquer instruções de comunicação ativas serão abortadas enquanto as alterações são descarregadas para o controlador.

Observe que o projeto original é exibido e o controlador está no modo de depuração.



Para aceitar as alterações

- Clique no ícone  Accept Changes.
- Observe que somente o ícone Alterar modo de operação está ativo agora e o controlador permanece no modo de depuração.



Observações:

Interrupções do usuário

As interrupções lhe permitem interromper seu programa com base em eventos definidos. Este capítulo contém informações sobre o uso de interrupções, as instruções sobre interrupções e a configuração de interrupções. O capítulo abrange os seguintes tópicos:

Tópico	Página
Informações sobre o uso de interrupções	297
Instruções sobre interrupções do usuário	300
Usando a função de interrupção selecionada em função do tempo (STI)	306
Configuração e status da interrupção selecionada em função do tempo (STI)	307
Usando a função de interrupção de entrada de evento (EII)	309

Para obter mais informações sobre interrupção HSC, consulte [Use o contador de alta velocidade e a chave de fim de curso programável na página 191](#).

Informações sobre o uso de interrupções

O objetivo desta seção é explicar algumas propriedades fundamentais das interrupções do usuário, inclusive:

- O que é uma interrupção?
- Quando se pode interromper uma operação do controlador?
- Prioridade de interrupções do usuário
- Configuração de interrupções
- Rotina de falha do usuário

O que é uma interrupção?

Uma interrupção é um evento que faz o controlador suspender a unidade de organização do programa (POU) que está sendo realizada, realizar uma POU diferente, e então retornar à POU suspensa no ponto em que foi suspensa. Os controladores Micro830, Micro850 e Micro870 suportam as seguintes interrupções do usuário:

- Rotina de falha do usuário
- Interrupções de evento (8)
- Interrupções do contador de alta velocidade (6)
- Interrupções selecionadas em função do tempo (4)
- Interrupções do módulo plug-in (5)

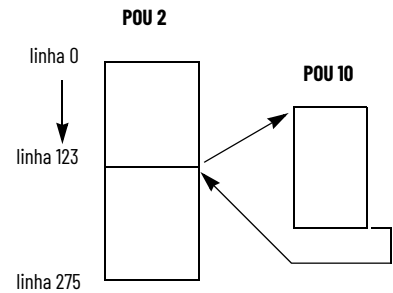
Uma interrupção deve ser configurada e habilitada para ser executada. Quando qualquer uma das interrupções é configurada (e habilitada) e ocorre em seguida, o programa do usuário:

1. suspende sua execução da POU atual,
2. executa uma POU predefinida com base em qual interrupção ocorreu, e
3. retorna à operação suspensa.

Exemplo de operação de interrupção

POU 2 é o programa principal de controle.
POU 10 é a rotina de interrupção.

- Um evento de interrupção ocorre na linha 123.
- POU 10 é executada.
- A execução de POU 2 é retomada imediatamente depois que POU 10 sofre a varredura.



Especificamente, se o programa do controlador está funcionando normalmente e ocorre um evento de interrupção:

1. O controlador pára seu funcionamento normal.
2. Determina qual interrupção ocorreu.
3. Vai imediatamente para o início da POU especificada por aquela interrupção do usuário.
4. Começa a executar a POU de interrupção do usuário (ou conjunto de POU/blocos de função se a POU especificada chamar um bloco de função subsequente).
5. Completa a POU.
6. Retoma o funcionamento normal a partir do ponto em que o programa do controlador foi interrompido

Quando se pode interromper uma operação do controlador?

Os controladores Micro830 permitem que as interrupções sejam realizadas em qualquer ponto de uma varredura de programa. Use as instruções UID/UIE para proteger o bloco de programas que não deve ser interrompido.

Prioridade de interrupções do usuário

Quando ocorrem múltiplas interrupções, elas são realizadas com base em sua prioridade individual.

Quando ocorre uma interrupção e outra(s) interrupção(ões) já ocorreu(ram) mas não foi(ram) executada(s), a nova interrupção é programada para execução com base em sua prioridade relativa às outras interrupções pendentes. No próximo ponto do tempo em que uma interrupção possa ser realizada, todas as interrupções são executadas na sequência da mais alta prioridade para a mais baixa prioridade.

Se uma interrupção ocorre enquanto uma interrupção de prioridade mais baixa está sendo executada, a rotina de interrupção que está sendo executada é suspensa, e a interrupção de prioridade mais alta é executada. Em seguida, a interrupção de prioridade mais baixa tem permissão para terminar antes de se retornar ao processamento normal.

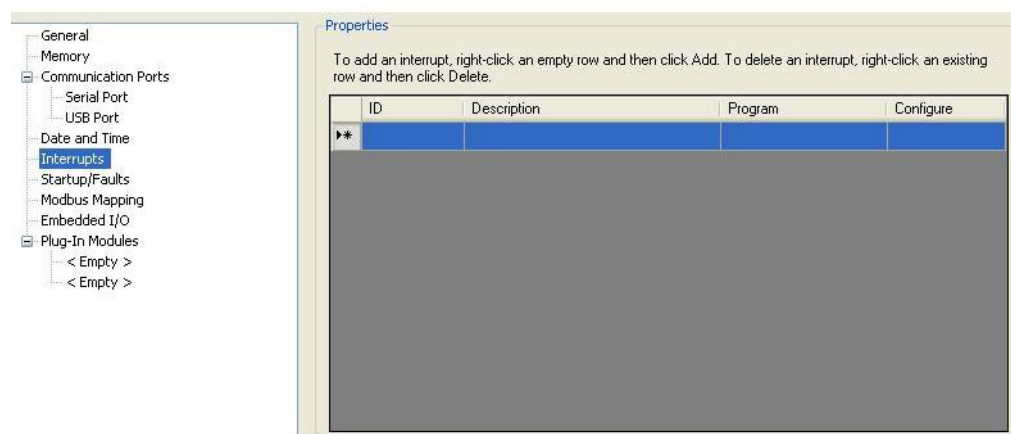
Se uma interrupção ocorre enquanto uma interrupção de prioridade mais alta está sendo em execução e o bit pendente foi definido para a interrupção de prioridade mais baixa, a rotina de interrupção que está sendo executada continuará até terminar. Em seguida, a interrupção de prioridade mais baixa é executada antes de se retornar ao processamento normal.

Tabela 58 - As prioridades da mais alta à mais baixa são

Rotina de falha do usuário	Prioridade mais alta
Interrupção de evento0	
Interrupção de evento1	
Interrupção de evento2	
Interrupção de evento3	
Interrupção do contador de alta velocidade0	
Interrupção do contador de alta velocidade1	
Interrupção do contador de alta velocidade2	
Interrupção do contador de alta velocidade3	
Interrupção do contador de alta velocidade4	
Interrupção do contador de alta velocidade5	
Interrupção de evento4	
Interrupção de evento5	
Interrupção de evento6	
Interrupção de evento7	
Interrupção temporizada selecionável0	
Interrupção temporizada selecionável1	
Interrupção temporizada selecionável2	
Interrupção temporizada selecionável3	
Interrupção de módulo plug-in0, 1, 2, 3, 4	Prioridade mais baixa

Configuração de interrupções do usuário

As interrupções do usuário podem ser configuradas e definidas como auto-iniciadas a partir da janela de interrupções.



Rotina de falha do usuário

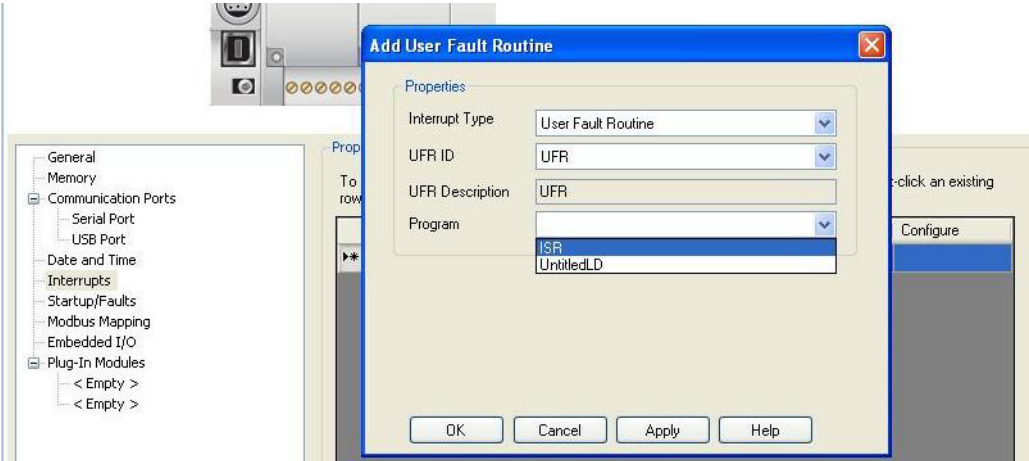
A rotina de falha do usuário lhe oferece a opção de realizar a limpeza antes de um encerramento do controlador, quando ocorre uma falha do usuário específica. A rotina de falha é executada quando ocorre quando falha do usuário. A rotina de falha não é executada para falhas que não são do usuário.

O controlador entra em modo de falha depois que uma rotina de falha do usuário é executada, e a execução do programa do usuário para.

Criando uma sub-rotina de falha do usuário

Para usar a sub-rotina de falha do usuário:

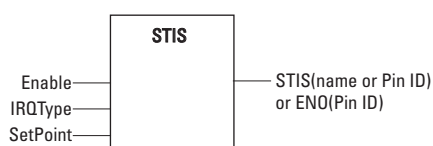
- 1. Crie uma POU.
- 2. Na janela User Interrupt Configuration, configure essa POU como uma rotina de falha do usuário.



Instruções sobre interrupções do usuário

Instrução	Usado para:	Página
STIS - Início com tempo selecionável	Use a instrução STIS (Início de interrupção selecionada em função do tempo) para o início do temporizador STI a partir do programa de controle, em vez de iniciar automaticamente.	301
UID - Desabilitar interrupção do usuário	Use as instruções Desabilitar interrupção do usuário (UID) e Habilitar interrupção do usuário (UIE) para criar zonas na quais as interrupções do usuário não podem ocorrer.	301
UIE - Habilitar interrupção do usuário		303
UIF - Fluxo de interrupção do usuário	Use a instrução UIF para remover do sistema interrupções pendentes selecionadas.	304
UIC - Remoção de interrupção do usuário	Use esta função para remover um bit de perda de interrupção para a interrupção do usuário selecionada.	305

STIS - Início selecionado em função do tempo



STIo é usada neste documento para definir como STIS funciona.

Tabela 59 - Parâmetros STIS

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição do parâmetro
Habilitar	Entrada	BOOL	Habilitar função. Quando Habilitar = TRUE, a função é realizada.. Quando Habilitar = FALSE, a função não é realizada.
IRQType	Entrada	UDINT	Use a DWORD definida por STI IRQ_STIO, IRQ_STI1, IRQ_STI2, IRQ_STI3
SetPoint	Entrada	UINT	O valor do tempo do intervalo de interrupção no temporizador do usuário em milissegundos. Quando SetPoint = 0, STI está desabilitado. Quando SetPoint = 1 a 65535, STI está habilitado.
STIS ou ENO	Saída	BOOL	Status da linha (mesmo que Habilitar)

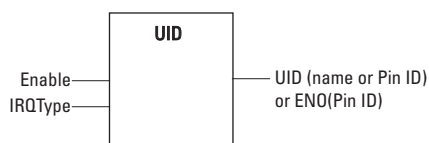
A instrução STIS pode ser utilizada para iniciar e parar a função STI ou para mudar o intervalo de tempo entre as interrupções do usuário STI. A instrução STI tem dois operandos:

- **IRQType** – É a ID da STI que um usuário deseja comandar.
- **SetPoint** – É a quantidade de tempo (em milissegundos) que deve passar antes de executar a interrupção do usuário selecionada em função do tempo. Um valor de zero desabilita a função STI. A faixa de tempo vai de 0 a 65.535 milissegundos.

A instrução STIS aplica o valor de referência especificado à função STI como segue (STIo é utilizado aqui como exemplo):

- Se um valor de referência de zero é especificado, o STI é desabilitado e STIo.Enable é removido (0).
- Se o STI está desabilitado (sem temporização) e um valor maior que 0 é inserido no valor de referência, o STI começa a temporizar o novo valor de referência e STIo.Enable é definido (1).
- Se o STI está temporizando neste momento e o valor de referência é mudado, o novo ajuste de parâmetro faz efeito imediatamente, recomeçando a partir de zero. O STI continua a temporizar até que alcance o novo valor de referência.

UID - Desabilitar interrupção do usuário



A instrução UID é utilizada para desabilitar interrupções do usuário selecionadas. A tabela abaixo mostra os tipos de interrupções com seus correspondentes bits de desabilitação:

Tabela 60 - Tipos de interrupções desabilitadas pela instrução UID

Tipo de interrupção	Elemento	Valor decimal	Bit correspondente
Módulo de encaixe	UPM4	8388608	bit 23
Módulo de encaixe	UPM3	4194304	bit 22
Módulo de encaixe	UPM2	2097152	bit 21
Módulo de encaixe	UPM1	1048576	bit 20
Módulo de encaixe	UPM0	524288	bit 19
STI - interrupção selecionada em função do tempo	STI3	262144	bit 18
STI - interrupção selecionada em função do tempo	STI2	131072	bit 17
STI - interrupção selecionada em função do tempo	STI1	65536	bit 16
STI - interrupção selecionada em função do tempo	STI0	32768	bit 15
EII - interrupção de entrada de evento	Evento 7	16384	bit 14
EII - interrupção de entrada de evento	Evento 6	8192	bit 13
EII - interrupção de entrada de evento	Evento 5	4096	bit 12
EII - interrupção de entrada de evento	Evento 4	2048	bit 11
HSC - contador de alta velocidade	HSC5	1024	bit 10
HSC - contador de alta velocidade	HSC4	512	bit 9
HSC - contador de alta velocidade	HSC3	256	bit 8
HSC - contador de alta velocidade	HSC2	128	bit 7
HSC - contador de alta velocidade	HSC1	64	bit 6
HSC - contador de alta velocidade	HSC0	32	bit 5
EII - interrupção de entrada de evento	Evento 3	16	bit 4
EII - interrupção de entrada de evento	Evento 2	8	bit 3
EII - interrupção de entrada de evento	Evento 1	4	bit 2
EII - interrupção de entrada de evento	Evento 0	2	bit 1
UFR - interrupção de rotina de falha do usuário	UFR	1	bit 0 (reservado)

Para desabilitar a(s) interrupção(ões):

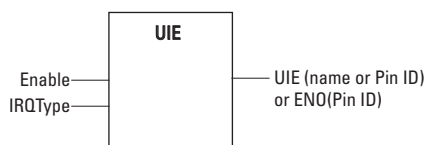
1. Selecione quais interrupções você deseja desabilitar.
2. Encontre o valor decimal para a(s) interrupção(ões) selecionadas.
3. Some os valores decimais se você selecionou mais de um tipo de interrupção.
4. Insira a soma na instrução UID.

Por exemplo, para desabilitar o evento EII 1 e o evento EII 3:

Evento EII 1 = 4, Evento EII 3 = 16

4 + 16 = 20 (digite este valor)

UIE - Habilitar interrupção do usuário



A instrução UIE é utilizada para habilitar interrupções do usuário selecionadas. A tabela abaixo mostra os tipos de interrupções com seus correspondentes bits de habilitação:

Tabela 61 - Tipos de interrupções habilitadas pela instrução UIE

Tipo de interrupção	Elemento	Valor decimal	Bit correspondente
Módulo de encaixe	UPM4	8388608	bit 23
Módulo de encaixe	UPM3	4194304	bit 22
Módulo de encaixe	UPM2	2097152	bit 21
Módulo de encaixe	UPM1	1048576	bit 20
Módulo de encaixe	UPM0	524288	bit 19
STI - interrupção selecionada em função do tempo	STI3	262144	bit 18
STI - interrupção selecionada em função do tempo	STI2	131072	bit 17
STI - interrupção selecionada em função do tempo	STI1	65536	bit 16
STI - interrupção selecionada em função do tempo	STI0	32768	bit 15
EII - interrupção de entrada de evento	Evento 7	16384	bit 14
EII - interrupção de entrada de evento	Evento 6	8192	bit 13
EII - interrupção de entrada de evento	Evento 5	4096	bit 12
EII - interrupção de entrada de evento	Evento 4	2048	bit 11
HSC - contador de alta velocidade	HSC5	1024	bit 10
HSC - contador de alta velocidade	HSC4	512	bit 9
HSC - contador de alta velocidade	HSC3	256	bit 8
HSC - contador de alta velocidade	HSC2	128	bit 7
HSC - contador de alta velocidade	HSC1	64	bit 6
HSC - contador de alta velocidade	HSC0	32	bit 5
EII - interrupção de entrada de evento	Evento 3	16	bit 4
EII - interrupção de entrada de evento	Evento 2	8	bit 3
EII - interrupção de entrada de evento	Evento 1	4	bit 2
EII - interrupção de entrada de evento	Evento 0	2	bit 1
		1	bit 0 (reservado)

Para habilitar a(s) interrupção(ões):

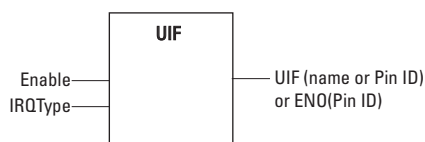
1. Selecione quais interrupções você deseja habilitar.
2. Encontre o valor decimal para a(s) interrupção(ões) selecionadas.
3. Some os valores decimais se você selecionou mais de um tipo de interrupção.
4. Insira a soma na instrução UIE.

Por exemplo, para habilitar o evento EII 1 e o evento EII 3:

Evento EII 1 = 4, Evento EII 3 = 16

4 + 16 = 20 (digite este valor)

UIF - Fluxo de interrupção do usuário



A instrução UIF é usada para limpar (remover do sistema interrupções pendentes) interrupções do usuário selecionadas. A tabela abaixo mostra os tipos de interrupções com seus correspondentes bits de rente:

Tabela 62 - Tipos de interrupções desabilitadas pela instrução UIF

Tipo de interrupção	Elemento	Valor decimal	Bit correspondente
Módulo de encaixe	UPM4	8388608	bit 23
Módulo de encaixe	UPM3	4194304	bit 22
Módulo de encaixe	UPM2	2097152	bit 21
Módulo de encaixe	UPM1	1048576	bit 20
Módulo de encaixe	UPM0	524288	bit 19
STI - interrupção selecionada em função do tempo	STI3	262144	bit 18
STI - interrupção selecionada em função do tempo	STI2	131072	bit 17
STI - interrupção selecionada em função do tempo	STI1	65536	bit 16
STI - interrupção selecionada em função do tempo	STI0	32768	bit 15
EII - interrupção de entrada de evento	Evento 7	16384	bit 14
EII - interrupção de entrada de evento	Evento 6	8192	bit 13
EII - interrupção de entrada de evento	Evento 5	4096	bit 12
EII - interrupção de entrada de evento	Evento 4	2048	bit 11
HSC - contador de alta velocidade	HSC5	1024	bit 10
HSC - contador de alta velocidade	HSC4	512	bit 9
HSC - contador de alta velocidade	HSC3	256	bit 8
HSC - contador de alta velocidade	HSC2	128	bit 7
HSC - contador de alta velocidade	HSC1	64	bit 6
HSC - contador de alta velocidade	HSC0	32	bit 5
EII - interrupção de entrada de evento	Evento 3	16	bit 4
EII - interrupção de entrada de evento	Evento 2	8	bit 3
EII - interrupção de entrada de evento	Evento 1	4	bit 2
EII - interrupção de entrada de evento	Evento 0	2	bit 1
UFR - interrupção de rotina de falha do usuário	UFR	1	bit 0 (reservado)

Para limpar a(s) interrupção(ões):

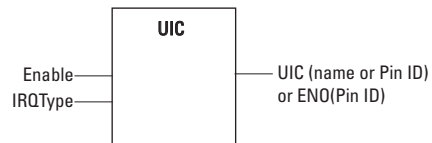
1. Selecione quais interrupções você deseja limpar.
2. Encontre o valor decimal para a(s) interrupção(ões) selecionadas.
3. Some os valores decimais se você selecionou mais de um tipo de interrupção.
4. Insira a soma na instrução UIF.

Por exemplo, para desabilitar o evento EII 1 e o evento EII 3:

Evento EII 1 = 4, Evento EII 3 = 16

4 + 16 = 20 (digite este valor)

UIC - Remoção de interrupção do usuário



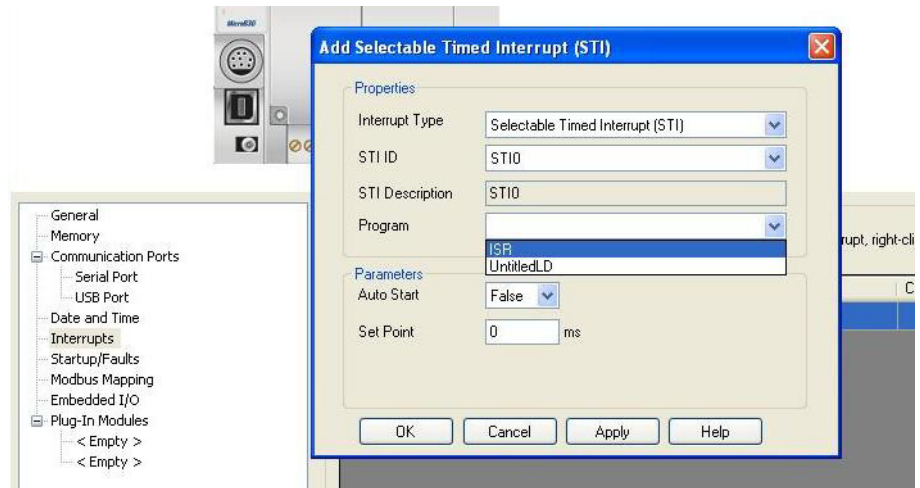
Esta função C remove um bit de perda de interrupção para a interrupção do usuário selecionada.

Tabela 63 - Tipos de interrupções desabilitadas pela instrução UIC

Tipo de interrupção	Elemento	Valor decimal	Bit correspondente
Módulo de encaixe	UPM4	8388608	bit 23
Módulo de encaixe	UPM3	4194304	bit 22
Módulo de encaixe	UPM2	2097152	bit 21
Módulo de encaixe	UPM1	1048576	bit 20
Módulo de encaixe	UPM0	524288	bit 19
STI - interrupção selecionada em função do tempo	STI3	262144	bit 18
STI - interrupção selecionada em função do tempo	STI2	131072	bit 17
STI - interrupção selecionada em função do tempo	STI1	65536	bit 16
STI - interrupção selecionada em função do tempo	STI0	32768	bit 15
EII - interrupção de entrada de evento	Evento 7	16384	bit 14
EII - interrupção de entrada de evento	Evento 6	8192	bit 13
EII - interrupção de entrada de evento	Evento 5	4096	bit 12
EII - interrupção de entrada de evento	Evento 4	2048	bit 11
HSC - contador de alta velocidade	HSC5	1024	bit 10
HSC - contador de alta velocidade	HSC4	512	bit 9
HSC - contador de alta velocidade	HSC3	256	bit 8
HSC - contador de alta velocidade	HSC2	128	bit 7
HSC - contador de alta velocidade	HSC1	64	bit 6
HSC - contador de alta velocidade	HSC0	32	bit 5
EII - interrupção de entrada de evento	Evento 3	16	bit 4
EII - interrupção de entrada de evento	Evento 2	8	bit 3
EII - interrupção de entrada de evento	Evento 1	4	bit 2
EII - interrupção de entrada de evento	Evento 0	2	bit 1
UFR - interrupção de rotina de falha do usuário	UFR	1	bit 0 (reservado)

Usando a função de interrupção selecionada em função do tempo (STI)

Configure a função STI a partir da janela Interrupt Configuration.



A interrupção selecionada em função do tempo (STI) fornece um mecanismo para resolver especificações de controle de tempo crítico. A STI é o mecanismo de disparo que lhe permite fazer a varredura ou resolver a lógica do programa de controle que é sensível ao tempo.

Seguem exemplos de onde se usaria uma STI:

- Aplicações de tipo PID, onde um cálculo deve ser realizado em um intervalo de tempo específico.
- Um bloco de lógica que precisa sofrer varreduras com maior frequência.

A maneira como uma STI é usada é tipicamente dirigida pelas solicitações/especificações da aplicação. Ela opera usando a seguinte sequência:

1. O usuário seleciona um intervalo de tempo.
2. Quando um intervalo válido é definido e a STI está adequadamente configurada, o controlador monitora o valor da STI.
3. Quando o período de tempo se esgota, a operação normal do controlador é interrompida.
4. O controlador faz então a varredura da lógica na POU da STI.
5. Quando a POU da STI é completada, o controlador retorna para onde estava antes da interrupção e continua sua operação normal.

Configuração e status da interrupção selecionada em função do tempo (STI)

Esta seção aborda a configuração e o gerenciamento de status da função STI.

Configuração da função STI

POU de programa STI

Este é o nome da unidade organizacional de programa (POU) que é executada imediatamente quando ocorre esta interrupção STI. Você pode escolher qualquer POU programada a partir da lista drop-down.

STI Auto Start (STIO.AS)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
AS – Autoinício	binário (bit)	somente leitura

O AS (Autoinício) é um bit de controle que pode ser utilizado no programa de controle. O bit de autoinício é configurado com o dispositivo de programação e armazenado como parte do programa do usuário. O bit de autoinício define automaticamente o bit de habilitação de interrupção temporizada STI (STIo.Enabled) quando o controlador entra em qualquer modo de execução.

Valor de referência STI milissegundos entre interrupções (STIO.SP)

Descrição de elemento secundário	Formato de dados	Faixa	Acesso ao programa do usuário
SP – Valor de referência ms	palavra (INT)	0 a 65,535	ler/gravar

Quando o controlador faz a transição para um modo de execução, o valor SP (referência em milissegundos) é carregado na STI. Se a STI está configurada corretamente, e habilitada, a POU na configuração da STI é executada nesse intervalo. Este valor pode ser mudado a partir do programa de controle usando a instrução STIS.



O valor mínimo não pode ser menor que o tempo necessário para fazer a varredura da POU da STI mais a latência da interrupção.

Informações de status da função STI

Os bits de status da função STI podem ser monitorados tanto no programa do usuário como no software Connected Components Workbench, no modo de depuração.

Execução de interrupção do usuário STI (STIO.EX)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
EX – Interrupção de execução do usuário	binário (bit)	somente leitura

O bit EX (Interrupção de execução do usuário) é definido sempre que o mecanismo STI completa a temporização e o controlador está fazendo a varredura da POU da STI. O bit EX é removido quando o controlador completa o processamento da sub-rotina STI.

O bit EX da STI pode ser utilizado no programa de controle como uma lógica condicional para detectar se uma interrupção STI está sendo executada.

Habilitar interrupção do usuário STI (STIO.Enabled)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
Habilitado - Habilitar interrupção do usuário	binário (bit)	somente leitura

O bit Habilitar interrupção do usuário é utilizado para indicar o status da STI de habilitar ou desabilitar.

Perda de interrupção do usuário STI (STIO.LS)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
LS - Perda de interrupção do usuário	binário (bit)	ler/gravar

O LS é um flag de status que indica que uma interrupção foi perdida. O controlador pode processar 1 condição de interrupção do usuário ativa e manter até 1 condição de interrupção do usuário pendente antes de energizar o bit perdido.

Este bit é energizado pelo controlador. A utilização, rastreamento da condição de perda, se necessários, dependem do programa de controle.

Interrupção do usuário pendente STI (STIO.PE)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
PE - Interrupção do usuário pendente	binário (bit)	somente leitura

O PE é um flag de status que representa uma interrupção que está pendente. Este bit de status pode ser monitorado ou utilizado para objetivos lógicos no programa de controle se você precisar determinar quando uma sub-rotina não pode ser executada imediatamente.

Este bit é definido e removido automaticamente pelo controlador. O controlador pode processar 1 condição de interrupção do usuário ativa e manter até 1 condição de interrupção do usuário pendente antes de energizar o bit perdido.

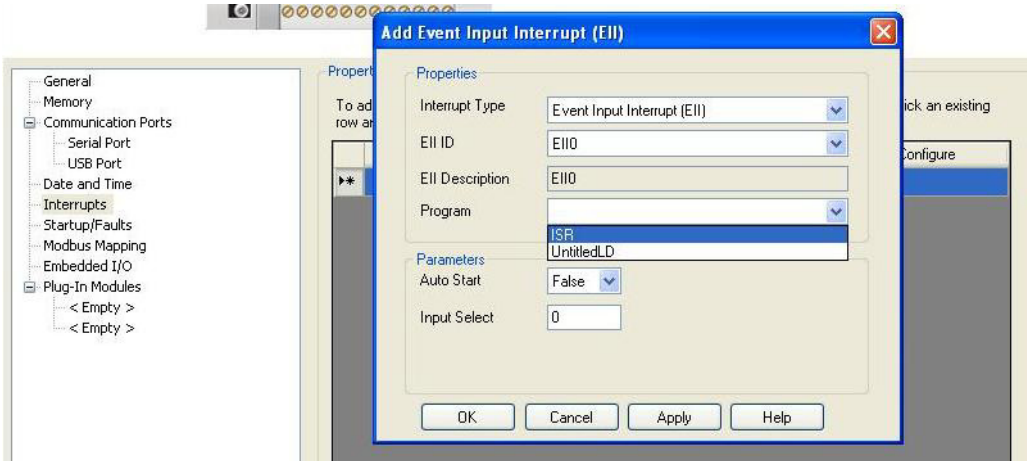
Usando a função de interrupção de entrada de evento (EII)

A EII (interrupção de entrada de evento) é um recurso que permite que o usuário faça a varredura de uma POU específica quando uma condição de entrada é detectada a partir de um dispositivo de campo.

EIIo é usada neste documento para definir como EII funciona.

Configure a borda de entrada EII a partir da janela Embedded I/O configuration.

Configure a EII a partir da janela Interrupt Configuration.



Configuração e status da função de interrupção de entrada de evento (EII)

Configuração da função EII

A função de interrupção de entrada de evento tem os seguintes parâmetros de configuração relacionados.

POU Programa EII

Este é o nome da unidade organizacional de programa (POU) que é executada imediatamente quando ocorre esta interrupção EII. Você pode escolher qualquer POU programada a partir da lista drop-down.

EII Autoinício (EII0.AS)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
AS - Autoinício	binário (bit)	somente leitura

O AS (Auto-início) é um bit de controle que pode ser utilizado no programa de controle. O bit de auto-início é configurado com o dispositivo de programação e armazenado como parte do programa do usuário. O bit de auto-início define automaticamente o bit habilitado de interrupção de usuário de evento quando o controlador entra em qualquer modo de execução.

EII Seleção de entrada (EII0.IS)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
IS - Seleção de entrada	palavra (INT)	somente leitura

O parâmetro IS (seleção de entrada) é utilizado para configurar cada EII para uma entrada específica no controlador. Entradas válidas são de 0 a N, onde N é 15 ou a ID de entrada máxima, o que for menor.

Este parâmetro é configurado com o dispositivo de programação e não pode ser modificado a partir do programa de controle.

Informações de status da função EII

Os bits de status da função EII podem ser monitorados tanto no programa do usuário como no software Connected Components Workbench, no modo de depuração.

EII Execução de interrupção do usuário (EII0.EX)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
EX - Interrupção de execução do usuário	binário (bit)	somente leitura

O bit EX (Interrupção de execução do usuário) é definido sempre que o mecanismo EII detecta uma entrada válida e o controlador está fazendo a varredura da POU da EII. O mecanismo EII remove o bit EX quando o controlador completa seu processamento da sub-rotina EII.

O bit EX da EII pode ser utilizado no programa de controle como uma lógica condicional para detectar se uma interrupção EII está sendo executada.

EII Habilitação de interrupção de usuário (EII0.Enabled)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
Habilitado - Habilitar interrupção do usuário	binário (bit)	somente leitura

O bit habilitado (Habilitar interrupção do usuário) é utilizado para indicar o status da EII de habilitar ou desabilitar.

EII Perda da interrupção do usuário (EII0.LS)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
LS - Perda de interrupção do usuário	binário (bit)	ler/gravar

LS (perda da interrupção do usuário) é um flag de status que representa uma interrupção que foi perdida. O controlador pode processar 1 condição de interrupção do usuário ativa e manter até 1 condição de interrupção do usuário pendente antes de energizar o bit perdido.

Este bit é energizado pelo controlador. A utilização ou o rastreamento da condição de perda, se necessário, depende do programa de controle.

ElI Interrupção do usuário pendente (ElI0.PE)

Descrição de elemento secundário	Formato de dados	Acesso ao programa do usuário
PE - Interrupção do usuário pendente	binário (bit)	somente leitura

O PE (interrupção do usuário pendente) é um flag de status que representa uma interrupção que está pendente. Este bit de status pode ser monitorado ou utilizado para objetivos lógicos no programa de controle se você precisar determinar quando uma sub-rotina não pode ser executada imediatamente.

Este bit é definido e removido automaticamente pelo controlador.

O controlador pode processar 1 condição de interrupção do usuário ativa e manter até 1 condição de interrupção do usuário pendente antes de energizar o bit perdido.

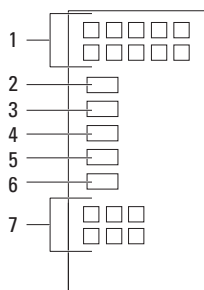
Observações:

Localização de falhas

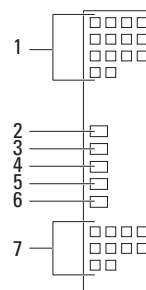
Indicadores de status no controlador

Micro830 Controladores

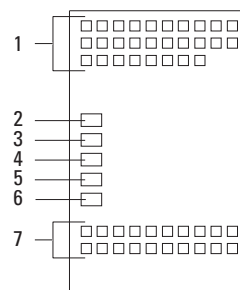
Controladores de 10/16 pontos



Controladores de 24 pontos

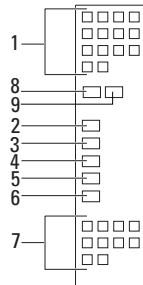


Controladores de 48 pontos

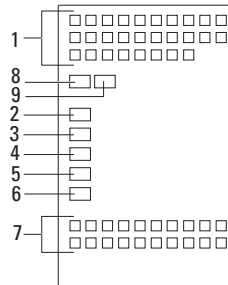


Micro850 Controladores

Controladores de 24 pontos

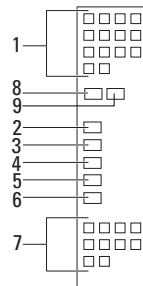


Controladores de 48 pontos



Controladores Micro870

Controladores de 24 pontos



Indicador de status, descrição

	Descrição	Estado	Indica
1	Status de entrada	Desligado	A entrada não está energizada
		On	A entrada está energizada (status do terminal).
2	Status de alimentação	Desligado	Sem alimentação de entrada, ou condição de erro de alimentação
		Verde	Energizado
3	Status de operação	Desligado	Não executando o programa do usuário
		Verde	Executando o programa do usuário em modo de operação.
		Verde intermitente	Backup de memória/restauração em andamento
4	Status de falha	Desligado	Nenhuma falha detectada.
		Vermelho	Falha não recuperável que exige desligar e religar.
		Vermelho intermitente	Falha recuperável.
5	Status de force	Desligado	Nenhuma condição de força está ativa.
		Âmbar	Condições de força estão ativas.
6	Status de comunicação serial	Desligado	Sem tráfego para RS-232/RS-485.
		Verde	Tráfego por RS-232/RS-485. O indicador somente pisca quando está transmitindo dados. Ele não pisca quando está recebendo dados.
7	Status de saída	Desligado	A saída não está energizada.
		On	A saída está energizada (status lógico).
8	Status do módulo	Permanentemente apagado	Sem alimentação.
		Verde intermitente	Em espera.
		Verde sólido	Dispositivo operacional.
		Vermelho intermitente	Falha de advertência (falhas recuperáveis graves e de advertência).
		Vermelho sólido	Falha grave (falha não recuperável).
		Verde e vermelho intermitente	Autoteste. O dispositivo está realizando um autoteste de energização (POST). Durante o POST, o indicador de status da rede alterna entre verde intermitente e vermelho. A duração do autoteste depende do tamanho do projeto no controlador.

Indicador de status, descrição (Continuação)

	Descrição	Estado	Indica
9	Status da rede	Permanentemente apagado	Não energizado, sem endereço IP. O dispositivo está desenergizado, ou está energizado mas sem endereço IP.
		Verde intermitente	Sem conexão. Um endereço IP está configurado, mas nenhuma aplicação Ethernet está conectada.
		Verde sólido	Conectado. Pelo menos uma sessão Ethernet/IP está estabelecida.
		Vermelho intermitente	Tempo-limite de conexão (não implementado).
		Vermelho sólido	IP duplicado. O dispositivo detectou que seu endereço IP está sendo utilizado por outro dispositivo na rede. Este status é aplicável somente se o recurso de detecção (ACD) do endereço IP duplicado do dispositivo estiver habilitado.
		Verde e vermelho intermitente	Autoteste. O dispositivo está realizando um autoteste de energização (POST). Durante o POST, o indicador de status da rede alterna entre verde intermitente e vermelho. A duração do autoteste depende do tamanho do projeto no controlador.

Operação normal

Os indicadores POWER e RUN estão acesos. Se uma condição de força está ativa, o indicador FORCE acende e permanece aceso até que todas as forças sejam removidas.

Códigos de erro

Esta seção lista possíveis códigos de erro para o seu controlador, assim como as ações recomendadas para recuperação. As informações sobre a falha estão armazenadas no log de falhas, que pode ser acessado a partir da página de diagnóstico no Software Connected Components Workbench. O log de falhas contém breves informações sobre a última falha, e informações detalhadas sobre as últimas 10 falhas não recuperáveis que ocorreram.

Se um erro persistir após a realização da ação recomendada, entre em contato com seu representante de suporte técnico local Rockwell Automation. Para obter informações de contato, acesse rok.auto/support.

Tipos de falha

Há dois tipos básicos de falhas que podem ocorrer:

- Recuperável – Uma falha recuperável pode ser removida sem precisar desligar e religar o controlador. O LED de falha pisca em vermelho quando ocorre uma falha recuperável.
- Não recuperável – Uma falha não recuperável exige que o controlador seja desligado e religado antes de remover a falha. Depois que o controlador foi desligado e religado ou reiniciado, verifique o log de falhas na página de diagnóstico do Software Connected Components Workbench, e então remova a falha. O LED de falha fica fixo em vermelho quando ocorre uma falha não recuperável.

Tabela 64 - Lista de códigos de erro para controladores Micro800

Código de erro	Tipo de falha	Descrição	Ação recomendada
OxF000	Recuperável	O controlador sofreu um reset inesperado por causa de um ambiente ruidoso ou uma falha de hardware interno. Se a variável do sistema _SYSVA_USER_DATA_LOST estiver definida, o controlador é capaz de recuperar o programa do usuário, mas os dados do usuário são removidos. Se não, o programa do controlador Micro800 é removido.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Verifique a fiação para eliminar qualquer ruído.
OxF001	Recuperável	O programa do controlador foi removido. Isso aconteceu porque: <ul style="list-style-type: none"> uma falha de energia ocorreu durante o download do programa ou transferência de dados a partir do módulo de memória. o cabo foi removido do controlador durante o download do programa. o teste de integridade RAM falhou. 	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Transfira o programa usando o utilitário de recuperação do módulo de memória.
OxF002	Não recuperável	O watchdog do hardware do controlador foi ativado. O tempo-limite de watchdog do hardware ocorre se a varredura do programa leva mais que 3 segundos. Se a variável do sistema _SYSVA_USER_DATA_LOST estiver definida, o controlador é capaz de recuperar o programa do usuário, mas os dados do usuário são removidos. Se não, o programa do controlador Micro800 é removido.	Consulte Ações corretivas para falhas não recuperáveis na página 321 .
OxF003	Recuperável	Ocorreu um dos seguintes: <ul style="list-style-type: none"> O hardware do módulo de memória está com falha. A conexão do módulo de memória está com falha. O módulo de memória era incompatível com a versão do firmware do controlador Micro800. 	Realize uma das seguintes ações: <ul style="list-style-type: none"> Remova o módulo de memória e reconecte-o. Obtenha um novo módulo de memória. Consulte Ações corretivas para falhas recuperáveis na página 321. Faça o upgrade da versão do firmware do controlador Micro800 para que seja compatível com o módulo de memória. Para obter informações sobre a compatibilidade da versão do firmware, acesse rok.auto/pcdc.
OxF004	Recuperável	Uma falha ocorreu durante a transferência de dados do módulo de memória.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Tente realizar a transferência de dados novamente. Se o erro persistir, substitua o módulo de memória. Para falha de RTC incorporado, reinicie o controlador.
OxF005	Recuperável	O programa do usuário falhou em uma verificação de integridade enquanto o controlador Micro800 estava no modo de operação.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Verifique a fiação.
OxF006	Recuperável	O programa do usuário é incompatível com a versão do firmware do controlador Micro800.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Contate o seu representante local de suporte técnico Rockwell Automation.
OxF010	Recuperável	O programa do usuário contém uma função/bloco de funções que não é suportado pelo controlador Micro800.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Contate o seu representante local de suporte técnico Rockwell Automation.
OxF014	Recuperável	Ocorreu um erro de memória do módulo de memória.	<ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Re programe o módulo de memória. Se o erro persistir, substitua o módulo de memória.
OxF015	Não recuperável	Ocorreu um erro de software inesperado.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas não recuperáveis na página 321. Verifique a fiação.
OxF016	Não recuperável	Ocorreu um erro de hardware inesperado.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas não recuperáveis na página 321. Verifique a fiação.

Tabela 64 - Lista de códigos de erro para controladores Micro800 (Continuação)

Código de erro	Tipo de falha	Descrição	Ação recomendada
0xF017	Não recuperável	Um erro inesperado de software ocorreu devido a uma interrupção inesperada do hardware. Se a variável do sistema _SYSVA_USER_DATA_LOST estiver definida, o controlador é capaz de recuperar o programa do usuário, mas os dados do usuário são removidos. Se não, o programa do controlador Micro800 é removido.	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas não recuperáveis na página 321 . • Verifique a fiação.
0xF018	Não recuperável	Um erro inesperado de software ocorreu devido a uma falha de comunicação SPI. Se a variável do sistema _SYSVA_USER_DATA_LOST estiver definida, o controlador é capaz de recuperar o programa do usuário, mas os dados do usuário são removidos. Se não, o programa do controlador Micro800 é removido.	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas não recuperáveis na página 321 . • Verifique a fiação.
0xF019	Não recuperável	Um erro de software inesperado ocorreu devido à memória ou outros problemas de recurso do controlador.	Consulte Ações corretivas para falhas não recuperáveis na página 321 .
0xF01A	Recuperável	O controlador sofreu um reset inesperado durante a alteração de modo de operação (RMC) por causa de um ambiente ruidoso ou uma falha de hardware interno. Se a variável do sistema _SYSVA_USER_DATA_LOST estiver definida, o controlador é capaz de recuperar o programa do usuário, mas os dados do usuário são removidos. Se não, o programa do controlador Micro800 é removido.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF020	Recuperável	O hardware de base com falha é incompatível com a versão do firmware do controlador Micro800.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF021	Recuperável	A configuração de E/S no programa do usuário é inválida ou não existe no controlador Micro800.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF022	Recuperável	O programa do usuário no módulo de memória é incompatível com a versão do firmware do controlador Micro800.	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Substitua o módulo de memória.
0xF023	Não recuperável	O programa do controlador foi removido. Isso aconteceu porque: • uma falha de energia ocorreu durante o download do programa ou transferência a partir do módulo de memória. • O teste de integridade de atualização falhou (somente Micro810).	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas não recuperáveis na página 321 . • Transfira ou faça download do programa.
0xF030 0xF031 0xF032 0xF033	Recuperável	Informações de desenergização na memória persistente podem não ter sido gravados corretamente devido a um ambiente com ruído ou uma falha interna de hardware. Se a variável do sistema _SYSVA_USER_DATA_LOST estiver definida, o controlador é capaz de recuperar o programa do usuário, mas os dados do usuário são removidos. Se não, o programa do controlador Micro800 é removido.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF050	Recuperável	A configuração de E/S incorporadas no programa do usuário é inválida.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF100	Recuperável	Há um erro de configuração geral detectado na configuração de movimento da que se fez download do software Connected Components Workbench, como um número de eixo ou intervalo de execução de movimento sendo configurado fora da faixa.	Realize as seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Corrija a configuração dos eixos no programa do usuário.
0xF110	Recuperável	Há um recurso de movimento faltando, como a variável Motion_DIAG não definida.	Realize as seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Corrija a configuração dos eixos no programa do usuário.

Tabela 64 - Lista de códigos de erro para controladores Micro800 (Continuação)

Código de erro	Tipo de falha	Descrição	Ação recomendada
0xF12z ⁽¹⁾	Recuperável	A configuração de movimento para o eixo z não pode ser suportada por este modelo de controlador, ou a configuração do eixo tem algum conflito de recurso com algum outro eixo de movimento que não foi configurado previamente.	Realize as seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Remova todos os eixos e reconfigure o movimento com a orientação do manual do usuário.
0xF15z ⁽¹⁾	Recuperável	Há um erro na lógica do motor de movimento (problema de lógica do firmware ou colisão de memória) para um eixo detectado durante a operação cíclica do motor de movimento. Uma razão possível pode ser um erro de memória/dados no motor de movimento.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF210	Recuperável	O terminador de expansion I/O está faltando.	Realize as seguintes ações: 1. Desligue o controlador. 2. Conecte o terminador de expansion I/O no último módulo de expansion I/O no sistema. 3. Ligue o controlador. 4. Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF230	Recuperável	O número máximo de módulos de expansion I/O foi excedido.	Realize as seguintes ações: 1. Desligue o controlador. 2. Certifique-se de que o número de módulos de expansion I/O não é maior que quatro. 3. Ligue o controlador. 4. Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF250	Recuperável	Há um erro não recuperável e o módulo de expansion I/O não pôde ser conectado.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF26z ⁽²⁾	Recuperável	Uma falha mestre de expansion I/O foi detectada no sistema.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xF27z ⁽²⁾	Recuperável	Uma falha de comunicação não recuperável ocorreu no módulo de expansion I/O.	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Substitua o módulo com o slot número z.
0xF28z ⁽²⁾	Recuperável	Erro de baudrate de expansion I/O.	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Substitua o módulo com o slot número z.
0xF29z ⁽²⁾	Recuperável	Uma falha de módulo foi detectada em seu módulo de expansion I/O.	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Substitua o módulo com o slot número z.
0xF2Az ⁽²⁾	Recuperável	Falha de alimentação de expansion I/O	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Substitua o módulo com o slot número z.
0xF2Bz ⁽²⁾	Recuperável	Falha de configuração expansion I/O.	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Corrija a configuração do módulo de expansion I/O no programa do usuário para que combine com a configuração real de hardware. • Verifique a operação e as condições do módulo de expansion I/O. • Substitua o módulo de expansion I/O.
0xF300	Recuperável	O módulo de memória está presente, mas está vazio e é necessária uma operação de restauração.	Realize as seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Verifique para se certificar de que há um projeto válido no módulo de memória. • Faça o download de um programa do usuário e use a função de backup para o módulo de memória.
0xF301	Recuperável	O projeto do módulo de memória não é compatível com o controlador.	Realize uma das seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Verifique para se certificar de que há um programa de usuário com um controlador que tem o catálogo correto do controlador configurado. • Faça o download de um programa do usuário e use a função de backup para o módulo de memória.

Tabela 64 - Lista de códigos de erro para controladores Micro800 (Continuação)

Código de erro	Tipo de falha	Descrição	Ação recomendada
0xF302	Recuperável	A senha não é igual entre o módulo de memória e o controlador. Aplica-se somente ao controlador Micro820 quando o LCD remoto realiza a operação de restauração. Esta falha não se aplica à revisão do firmware do controlador Micro800 10 e posteriores.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Verifique para se certificar de que o programa de usuário no módulo de memória tem a senha correta. Faça o download de um programa do usuário com uma senha e use a função de backup para o módulo de memória. Use o software Connected Components Workbench para inserir a senha correta no controlador e realizar novamente a operação de restauração.
0xF303	Recuperável	O módulo de memória não está presente e é necessária uma operação de restauração.	Verifique para se certificar de que o módulo de memória está presente.
0xF0A ⁽³⁾	Recuperável	O módulo de E/S plug-in sofreu um erro durante a operação.	Realize as seguintes ações: <ul style="list-style-type: none"> Verifique a operação e as condições do módulo de E/S plug-in. Consulte Ações corretivas para falhas recuperáveis na página 321.
0xF0B ⁽³⁾	Recuperável	A configuração do módulo de E/S plug-in não combina com a configuração de E/S real detectada.	Realize uma das seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Corrija a configuração do módulo de E/S plug-in no programa do usuário para que combine com a configuração real de hardware. Verifique a operação e as condições do módulo de E/S plug-in. Substitua o módulo de E/S plug-in.
0xF0D ⁽³⁾	Recuperável	Quando a alimentação foi aplicada ao módulo de E/S plug-in ou o módulo de E/S plug-in foi removido, ocorreu um erro de hardware.	Realize as seguintes ações: <ol style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Corrija a configuração do módulo de E/S plug-in no programa do usuário. Construa e faça download do programa usando o software Connected Components Workbench. Coloque o controlador Micro800 no modo de operação.
0xF0E ⁽³⁾	Recuperável	A configuração do módulo de E/S plug-in não combina com a configuração de E/S real detectada.	Realize as seguintes ações: <ol style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Corrija a configuração do módulo de E/S plug-in no programa do usuário. Construa e faça download do programa usando o software Connected Components Workbench. Coloque o controlador Micro800 no modo de operação.
0xF830	Recuperável	Ocorreu um erro na configuração EII.	Realize as seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Revise e altere a configuração EII nas propriedades do controlador Micro800.
0xF840	Recuperável	Ocorreu um erro na configuração HSC.	Realize as seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Revise e altere a configuração EII nas propriedades do controlador Micro800.
0xF850	Recuperável	Ocorreu um erro na configuração STI.	Realize as seguintes ações: <ul style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Revise e altere a configuração EII nas propriedades do controlador Micro800.
0xF860	Recuperável	Ocorreu uma sobrecontagem de dados. Um erro de sobrecontagem de dados é gerado quando a execução da lógica ladder, do texto estruturado ou do diagrama de blocos de funções (FBD) encontra uma divisão por zero.	Realize as seguintes ações: <ol style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Corrija o programa para assegurar que não há sobrecontagem de dados. Construa e faça download do programa usando o software Connected Components Workbench. Coloque o controlador Micro800 no modo de operação.
0xF870	Recuperável	Um endereço de índice estava fora do espaço de dados.	Realize as seguintes ações: <ol style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Corrija o programa para garantir que não há índice usado para acessar um elemento de vetor além dos limites do vetor. Construa e faça download do programa usando o software Connected Components Workbench. Coloque o controlador Micro800 no modo de operação.
0xF0878	Recuperável	Um índice usado para acessar um bit está além dos limites do tipo de dados no qual é utilizado.	Realize as seguintes ações: <ol style="list-style-type: none"> Consulte Ações corretivas para falhas recuperáveis na página 321. Corrija o programa para garantir que não há índice usado para acessar um bit além dos limites do tipo de dados. Construa e faça download do programa usando o software Connected Components Workbench. Coloque o controlador Micro800 no modo de operação.

Tabela 64 - Lista de códigos de erro para controladores Micro800 (Continuação)

Código de erro	Tipo de falha	Descrição	Ação recomendada
0xF880	Recuperável	Ocorreu um erro de conversão de dados.	Realize as seguintes ações: 1. Consulte Ações corretivas para falhas recuperáveis na página 321 . 2. Corrija o programa para assegurar que não há erro de conversão de dados. 3. Construa e faça download do programa usando o software Connected Components Workbench. 4. Coloque o controlador Micro800 no modo de operação.
0xF888	Recuperável	A pilha de chamadas do controlador não é compatível com a sequência de chamadas a blocos de funções no atual projeto. Um excesso de blocos está dentro de outro bloco.	Realize as seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Mude o projeto para reduzir a quantidade de blocos sendo chamados dentro de um bloco.
0xF898	Recuperável	Ocorreu um erro na configuração da interrupção do usuário para o módulo de E/S plug-in.	Realize as seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Corrija a configuração da interrupção do usuário para o módulo de E/S plug-in no programa do usuário para que combine com a configuração real de hardware.
0xF8A0	Recuperável	Os parâmetros TOW são inválidos.	Realize as seguintes ações: 1. Consulte Ações corretivas para falhas recuperáveis na página 321 . 2. Corrija o programa para assegurar que não há parâmetros inválidos. 3. Construa e faça download do programa usando o software Connected Components Workbench. 4. Coloque o controlador Micro800 no modo de operação.
0xF8A1	Recuperável	Os parâmetros DOY são inválidos.	Realize as seguintes ações: 1. Consulte Ações corretivas para falhas recuperáveis na página 321 . 2. Corrija o programa para assegurar que não há parâmetros inválidos. 3. Construa e faça download do programa usando o software Connected Components Workbench. 4. Coloque o controlador Micro800 no modo de operação.
0xFFzz ⁽⁴⁾	Recuperável	Ocorreu uma falha criada pelo usuário do software Connected Components Workbench.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xD00F	Recuperável	Um tipo especial de hardware (por exemplo, E/S incorporada) foi selecionado na configuração do programa do usuário, mas não combina com a base real de hardware.	Consulte Ações corretivas para falhas recuperáveis na página 321 .
0xD011	Recuperável	O tempo de varredura do programa excedeu o valor de limite de tempo watchdog.	Realize as seguintes ações: • Consulte Ações corretivas para falhas recuperáveis na página 321 . • Determine se o programa ficou preso em uma malha e corrija o problema. Uma falha pode ocorrer se seu programa de texto estruturado contiver uma malha com o limite superior definido para o valor máximo da variável. Por exemplo, a variável é um USINT e o limite está definido para 255, ou a variável é um UINT e o limite está definido para 65535. Para corrigir a falha, realize o seguinte: 1. Corrija o programa para garantir que o limite superior não seja alcançado. Um método é usar um tipo de dado com um valor máximo maior. 2. Construa e faça download do programa usando o software Connected Components Workbench. 3. Coloque o controlador Micro800 no modo de operação. Se o seu programa é projetado para ter um tempo de varredura mais longo que 3 segundos, no programa do usuário, aumente o tempo-limite de watchdog que está configurado na variável do sistema _SYSVA_TCYWDG e então construa e descarregue o programa usando o software Connected Components Workbench.

(1) z indica a ID do eixo lógico. (0 a 3).

(2) z indica o número do slot da E/S de expansão. Se z=0, então o número do slot não pode ser identificado.

(3) z é o número do slot do módulo plug-in. Se z=0, então o número do slot não pode ser identificado.

(4) zz indica o último byte do número do programa. Somente número de programa até 0xFF podem ser exibidos. Para números de programa de 01x00 a 0xFFFF, somente o último byte é exibido.)

Ação corretiva para falhas recuperáveis e não recuperáveis

Ações corretivas para falhas recuperáveis

Realize as seguintes ações:

1. Grave opcionalmente o log de falhas do Software Connected Components Workbench.
2. Remova a falha recuperável usando o Software Connected Components Workbench.
3. Se o problema persistir, entre em contato com o suporte técnico com o log de falhas.

Ações corretivas para falhas não recuperáveis

Realize as seguintes ações:

1. Desligue e ligue seu controlador Micro800.
2. O controlador passará a uma falha recuperável. Grave opcionalmente o log de falhas do Software Connected Components Workbench.
3. Remova a falha recuperável usando o Software Connected Components Workbench.
4. Se o programa for perdido, construa e descarregue o seu programa usando o Software Connected Components Workbench.
5. Se o problema persistir, entre em contato com o suporte técnico com o log de falhas.

Recuperar um registro de falha

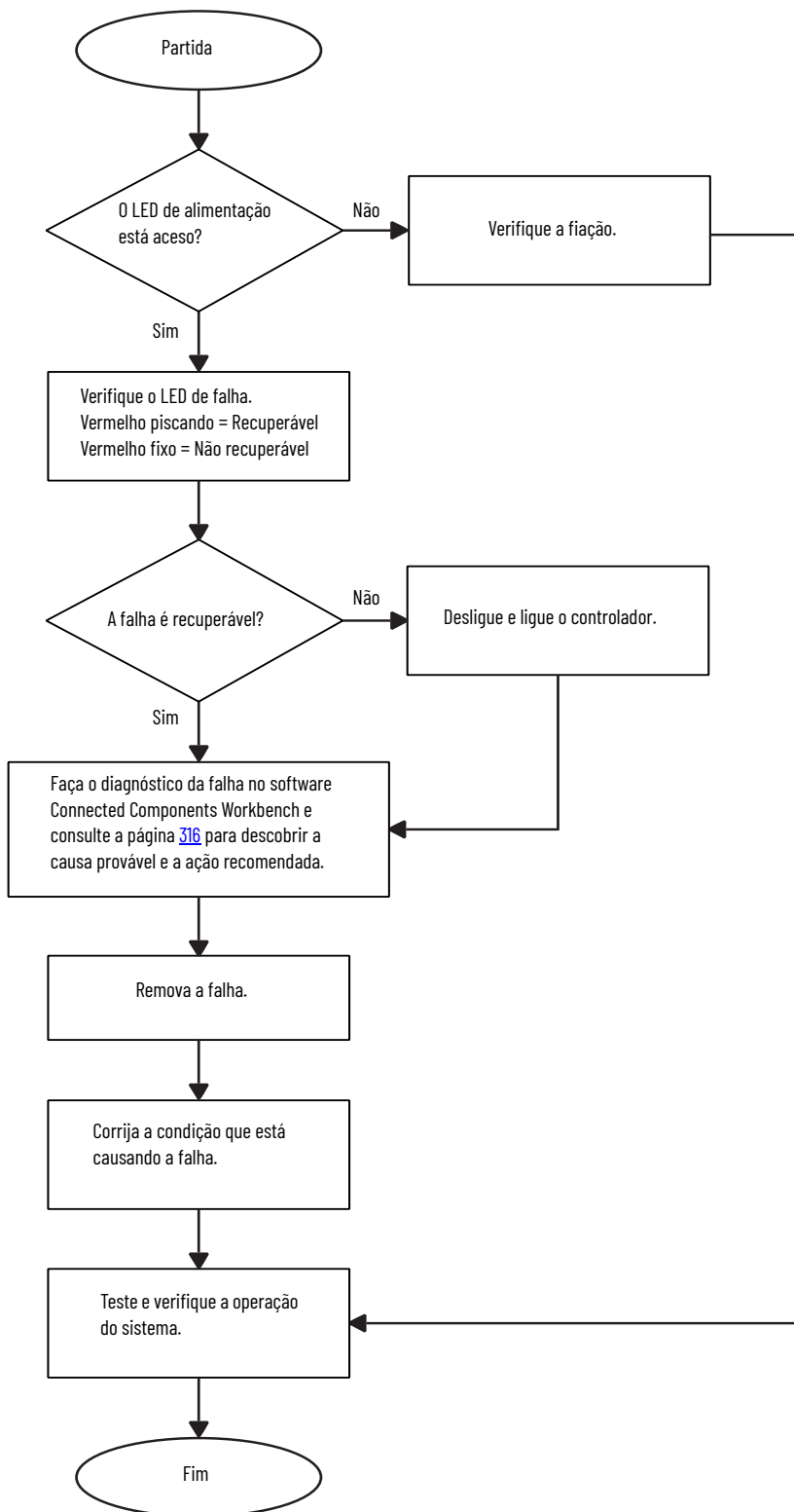
Você pode recuperar um log de falha para seu controlador usando o software Connected Components Workbench, versão 9 ou posterior.

Realize as seguintes ações:

1. Execute o software Connected Components Workbench.
2. Conecte ao controlador Micro800
3. Em Project Organizer, clique com o botão direito do mouse no controlador Micro800.
4. Selecione Diagnose > Fault.
A aba Fault Diagnostics é exibida.
5. Clique no botão Get Fault Log.
6. Salve o arquivo de registro de falha (.txt).

Modelo de recuperação de erro do controlador

Use o seguinte modelo de recuperação de erro para ajudá-lo a diagnosticar problemas de software e hardware no microcontrolador. O modelo traz dúvidas comuns que você pode questionar para ajudar a encontrar o problema em seu sistema. Consulte as páginas recomendadas no modelo para mais ajuda.



Ligar para a Rockwell Automation para obter assistência

Se você precisar entrar em contato com a Rockwell Automation ou seu distribuidor local para assistência, antes de ligar, tenha em mãos:

- tipo do controlador, carta de série, carta de revisão e número de firmware (FRN) do controlador
- status do indicador do controlador

Blocos de função PID

O bloco de função PID tem nomenclatura de parâmetros similar ao RSLogix 500 e é recomendado para usuários que já têm familiaridade com programação em RSLogix 500. O bloco de função IPIDCONTROLLER tem a vantagem de suportar o ajuste automático.

Tabela 65 - Comparação entre IPIDCONTROLLER e PID

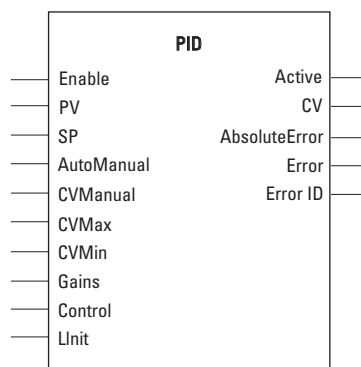
IPIDCONTROLLER	PID	Descrição
Parâmetros comuns		
Processo	PV	Realimentação de variável de processo
SetPoint	SP	Entrada de configuração
Saída	CV	Saída CV
Gains.DirectActing	Controle	Direção de controle do processo (resfriamento versus aquecimento).
Gains.ProportionalGain	Gains.Kc	Ganho do controlador tanto para P quanto para I
Gains.TimeIntegral	Gains.Ti	Valor integral de tempo para I
Gains.TimeDerivative	Gains.Td	Valor derivativo de tempo para D
Gains.DerivativeGain	Gains.FC	Uma constante de filtro maior torna a saída CV mais responsiva a erros. Age como um ganho derivativo.
AbsoluteError	AbsoluteError	Valor absoluto do erro
Parâmetros específicos de PID		
-	CVMin	Para limitar CV
	CVMax	Para limitar CV
	AutoManual	VERDADEIRO = Operação normal de PID. FALSO = Operação manual usando CVManual.
	CVManual	CV quando no modo manual
	Habilitar	VERDADEIRO = Iniciar execução com os parâmetros de entrada atuais. FALSO = CV é igual a zero.
	Ativo	VERDADEIRO = estado PID é em funcionamento. FALSO = estado PID é parado.
	Erro	VERDADEIRO = PID tem um erro. FALSO = PID não tem erros.
	ErrorID	ID do erro PID

Tabela 65 - Comparação entre IPIDCONTROLLER e PID (Continuação)

IPIDCONTROLLER	PID	Descrição
Parâmetros específicos de IPIDCONTROLLER		
Auto	-	VERDADEIRO = Operação normal de PID. FALSO = Saída rastreia a realimentação.
Realimentação		Realimentação do controle sendo aplicado ao processo. Geralmente é o CV do PID após a aplicação de quaisquer limites ou controle manual.
AutoTune		VERDADEIRO = ajuste automático. FALSO = sem ajuste automático.
ATParameters		Parâmetros de autoajuste
ATWarning		Advertência de ajuste automático
OutGains		Ganho do ajuste automático
Initialize		Usado para ajuste automático

Bloco de funções PID

Esse diagrama de bloco de funções mostra os argumentos no bloco de funções PID



A tabela a seguir explica os argumentos usados neste bloco de funções.

Tabela 66 - Argumentos do PID

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição
Habilitar	Entrada	BOOL	Habilitar instrução. VERDADEIRO = Iniciar execução com os parâmetros de entrada atuais. FALSO = CV é igual a zero.
PV	Entrada	REAL	Valor do processo. Este valor é tipicamente lido a partir de um módulo de entrada analógica. A unidade SI deve ser a mesma da configuração.
SP	Entrada	REAL	O valor do ponto de ajuste para o processo.
AutoManual	Entrada	BOOL	Seleção do modo auto ou manual: VERDADEIRO = Operação normal de PID. FALSO = Operação manual usando CVManual.
CVManual	Entrada	REAL	Entrada do valor de controle definida para operação em modo manual. A faixa válida para CVManual é: $CVMin < CVManual < CVMax$
CVMin	Entrada	REAL	Limite mínimo do valor de controle. Se $CV < CVMin$, então $CV = CVMin$. Se $CVMin > CVMax$, e ocorre um erro.
CVMax	Entrada	REAL	Limite máximo do valor de controle. Se $CV > CVMax$, então $CV = CVMax$. Se $CVMax < CVMin$, ocorre um erro.

Tabela 66 - Argumentos do PID (Continuação)

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição
Gains	Entrada	PID_GAINS	Ganhos de PID para controlador. Use o tipo de dado PID_GAINS para configurar o parâmetro de ganhos.
Controle	Entrada	BOOL	Direção de controle do processo: VERDADEIRO = Ação direta, como resfriar. FALSO = Ação reversa, como aquecer.
LInit	Entrada	BOOL	Reservado para uso futuro.
Ativo	Saída	BOOL	Status do controlador PID: VERDADEIRO = estado PID é em funcionamento. FALSO = estado PID é parado.
CV	Saída	REAL	A saída do valor de controle. Se ocorreu um erro, CV é 0.
AbsoluteError	Saída	REAL	Erro absoluto é a diferença entre o valor de processo (PV) e o valor de configuração (SV).
Erro	Saída	BOOL	Indica a existência de uma condição de erro. VERDADEIRO = PID tem um erro. FALSO = PID não tem erros.
ErrorID	Saída	USINT	Um número único que identifica o erro. Os erros são definidos nos códigos de erro PID.

Tabela 67 - Tipo de dados GAIN_PID

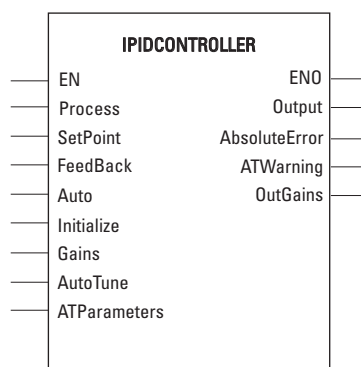
Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição
KC	Entrada	REAL	Ganho do controlador para PID. Proporcional e Integral são dependentes deste ganho ($\geq 0,0001$). Kc em aumento melhora o tempo de resposta, mas também aumenta o excesso e a oscilação do PID. Se Kc for inválido, ocorre um erro.
Ti	Entrada	REAL	Constante de tempo integral em segundos ($\geq 0,0001$). Ti em aumento diminui o excesso e a oscilação do PID. Se Ti for inválido, ocorre um erro.
TD+	Entrada	REAL	Constante de tempo derivativo em segundos ($\geq 0,0$). Quando Td é igual a 0, não há ação derivativa e PID se torna um controlador PI. Td em aumento reduz o excesso e remove a oscilação do controlador PID. Se Td for inválido, ocorre um erro.
FC	Entrada	REAL	Constante de filtro ($\geq 0,0$). O intervalo recomendado para FC é de 0 a 20. FC em aumento alivia a resposta do controlador PID. Se FC for inválido, ocorre um erro.

Tabela 68 - Códigos de erro do PID

Código de erro	Descrição
0	PID está funcionamento normalmente.
1	Kc é inválido.
2	Ti é inválido.
3	Td é inválido.
4	FC é inválido.
5	CVMin > CVMax, ou CVMax < CVMin
6	CVManual < CVMin CVManual é inválido.
7	CVManual > CVMax CVManual é inválido.

Bloco de funções IPIDCONTROLLER

Esse diagrama de bloco de funções mostra os argumentos no bloco de funções IPIDCONTROLLER.



A tabela a seguir explica os argumentos usados neste bloco de funções.

Tabela 69 - Argumentos do IPIDCONTROLLER

Parâmetro	Tipo de parâmetro	Tipo de dados	Descrição
EN	Entrada	BOOL	Habilitação de bloco de funções TRUE = Executar função. FALSE = Não executar função. Aplicável a programas de diagrama ladder.
Processo	Entrada	REAL	Valor do processo, que é o valor medido a partir da saída do processo.
SetPoint	Entrada	REAL	O valor do ponto de ajuste para o processo.
Realimentação	Entrada	REAL	Sinal de realimentação, que é o valor da variável de controle aplicada ao processo. Por exemplo, a realimentação pode ser a saída IPIDCONTROLLER.
Auto	Entrada	BOOL	Modos de operação do controlador PID: VERDADEIRO = Operação normal de PID. FALSO = Saída rastreia a realimentação.
Initialize	Entrada	BOOL	Uma mudança no valor (VERDADEIRO para FALSO ou FALSO para VERDADEIRO) faz com que o controlador elimine qualquer ganho proporcional durante aquele ciclo. Ele também inicializa sequências AutoTune.
Gains	Entrada	GAIN_PID	PID de ganhos para IPIDCONTROLLER. Use o tipo de dado GAIN_PID para definir os parâmetros para a entrada de ganhos.
AutoTune	Entrada	BOOL	VERDADEIRO = ajuste automático. FALSO = sem ajuste automático.
ATParameters	Entrada	AT_Param	Parâmetros de autoajuste Use o tipo de dado AT_Param para definir os parâmetros para a entrada ATParameters.
Saída	Saída	Real	Valor da saída a partir do controlador
AbsoluteError	Saída	Real	Erro absoluto (Processo - Referência) para o controlador.
ATWarnings	Saída	DINT	Advertência para a sequência de ajuste automático. Os valores possíveis são: 0 - Nenhum autotune realizado 1 = No modo de ajuste automático. 2 = Nenhum autotune realizado. -1 = Erro 1: Entrada automaticamente configurada como TRUE, o ajuste automático não é possível. -2 = Erro 2: Erro de ajuste automático, o ATDynamSet expirou.
OutGains	Saída	GAIN_PID	Ganhos calculados a partir das sequências de AutoTune. Use o tipo de dado GAIN_PID para definir a saída OutGains.
ENO	Saída	BOOL	Habilitar saída. Aplicável a programas de diagrama ladder.

Tabela 70 - Tipo de dados GAIN_PID

Parâmetro	Tipo	Descrição
DirectActing	BOOL	Tipos de ação: TRUE = Ação direta, saída se move na mesma direção que o erro. Isto é, o valor real do processo é maior que a configuração e a ação adequada do controlador é aumentar a saída. Por exemplo, resfriar. FALSE = Ação reversa, saída se move na direção oposta à do erro. Isto é, o valor real do processo é maior que a configuração e a ação adequada do controlador é diminuir a saída. Por exemplo, aquecer.
ProportionalGain	REAL	Ganho proporcional para PID ($\geq 0,0001$) Ganho proporcional para PID (P_Gain) Um ganho proporcional maior causa uma mudança maior na saída com base na diferença entre o PV (valor do processo medido) e SV (valor de referência). Quanto maior o ganho, mais rápido o erro diminui, mas isso pode resultar em instabilidade, como oscilações. Quanto menor o ganho, mais lentamente o erro diminui, mas o sistema fica mais estável e menos sensível a erros grandes. O P_Gain geralmente é o ganho mais importante a ajustar e o primeiro ganho a ajustar quando se está realizando o tuning.
TimeIntegral	REAL	Valor integral de tempo para PID ($\geq 0,0001$) Valor integral de tempo para PID Uma constante de tempo integral menor causa uma mudança mais rápida na saída com base na diferença entre o PV (valor do processo medido) e SV (valor de referência) integrado ao longo desse tempo. Uma constante de tempo integral menor diminui o erro de estado estável (erro quando SV não está sendo alterado), mas aumenta as chances de instabilidade como oscilações. Uma constante de tempo integral maior deixa mais lenta a resposta do sistema e o torna mais estável, mas PV se aproxima de SV a uma taxa mais lenta.
TimeDerivative	REAL	Valor derivativo de tempo para PID ($\geq 0,0$) Valor derivativo de tempo para PID (Td) Uma constante de tempo derivativo menor causa uma mudança mais rápida na saída com base na taxa de mudança da diferença entre o PV (valor do processo medido) e SV (valor de referência). Uma constante de tempo derivativo menor torna um sistema mais responsivo a mudanças repentinas em erro (SV é alterado), mas aumenta as chances de instabilidade como oscilações. Um tempo constante maior torna o sistema menos responsivo a mudanças repentinas no erro e o sistema é menos suscetível a ruído e mudanças de etapa no PV. O TimeDerivative (Td) está relacionado ao ganho derivativo, mas permite que a contribuição derivativa para o PID seja ajustada usando o tempo para que o tempo de amostra seja levado em consideração.
DerivativeGain	REAL	Ganho derivativo para PID ($\geq 0,0$) Ganho derivativo para PID (D_Gain) Um ganho derivativo maior causa uma mudança maior na saída com base na taxa de mudança da diferença entre o PV (valor do processo medido) e SV (valor de referência). Um ganho maior torna o sistema mais responsivo a mudanças repentinas no erro, mas aumenta as chances de instabilidade como oscilações. Um ganho menor torna o sistema menos responsivo a mudanças repentinas no erro e torna o sistema menos suscetível a ruído e mudanças de etapa no PV. Se o ganho derivativo for configurado em zero, ele desabilita a porção derivativa do PID.

Tabela 71 - Tipo de dados AT_Param

Parâmetro	Tipo	Descrição
Carga	REAL	Carregar parametro para ajuste automático. Este é o valor da saída quando se inicia o ajuste automático.
Diferença	REAL	Desvio para autotune. Este é o desvio padrão utilizado para avaliar a faixa de ruído necessária para AutoTune (faixa de ruído = $3 \times$ desvio) ⁽¹⁾
Etapa	REAL	Valor de etapa para AutoTune. Deve ser superior à faixa de ruído e menos que $\frac{1}{2}$ da carga.
ATDynamSet	REAL	Tempo de espera em segundos antes de abandonar o ajuste automático.
ATReset	BOOL	Determina se o valor da saída é zerado após uma sequência de AutoTune: VERDADEIRO = Restaurar saída para zero. FALSO - deixa a saída com o valor de carga

(1) O engenheiro da aplicação pode estimar o valor de ATParams.Deviation observando o valor de entrada do processo. Por exemplo, em um projeto que envolva controle de temperatura, se esta for estabilizada em torno de 22°C, e for observada uma flutuação de 21,7 a 22,5 °C, o valor de ATParams.Deviation será $(22,5 + 21,7)/2 = 0,4$.

Como fazer o ajuste automático

Antes de fazer o autotune, é necessário:

- Verificar se seu sistema está constante quando não há controle. Por exemplo, para controle de temperatura, o valor de processo deve permanecer em temperatura ambiente quando não há saída de controle.
- Configure o valor de referência como 0.
- Defina Auto Input como False.
- Defina o parâmetro Gain da seguinte maneira:

Tabela 72 - Valores do parâmetro GAIN

Parâmetro GAIN	Valor
DirectActing	De acordo com a operação: TRUE (por exemplo, resfriamento), ou FALSE (por exemplo, aquecimento)
DerivativeGain	0,5
ProportionalGain	0,0001
TimeIntegral	0,0001
TimeDerivative	0,0

- Defina o AT_Parameter da seguinte maneira:

Tabela 73 - Valores do AT_Parameter

Parâmetro AT	Recomendações
Carga	Toda 'Carga' fornece um valor de processo saturado por um período de tempo. Ajuste a carga do valor para o valor de processo saturado desejado. IMPORTANTE: Se a carga de 40 resulta em um valor de processo de 30°C por um período de tempo e você deseja ajustar seu sistema para 30°C, você deve definir a carga como 40.
Diferença	Este parâmetro tem um papel importante no processo autotune. O método de derivação deste valor é explicado posteriormente nesta seção. Não é necessário definir este parâmetro antes do autotuning. No entanto, se o desvio já for conhecido, ele pode ser definido primeiro.
Etapas	O valor da etapa deve ser entre 3*desvio e ½ carga. A etapa fornece um desvio para a carga durante o autotuning. Ela deve ser definida com um valor alto o suficiente para criar uma mudança significativa no valor do processo.
ATDynamSet	Defina este valor com um tempo razoavelmente longo para o processo autotune. Todo sistema é diferente, portanto, permita um tempo maior para um sistema com um valor de processo que leve mais tempo para reagir à mudança.
ATReset	Defina este parâmetro como TRUE para reinicializar a saída como zero após o processo de autotune ser concluído. Defina este parâmetro como FALSE para deixar a saída com o valor de carga após o processo de autotune ser concluído.

Durante o ajuste automático, o controlador definirá automaticamente o valor do processo como zero. Para realizar o autotune, siga as seguintes etapas:

1. Defina a entrada "Initialize" para "VERDADEIRA".
2. Defina a entrada "AutoTune" para "VERDADEIRA".
3. Aguarde que a entrada "Process" se estabilize ou chegue no regime permanente.
4. Observe a flutuação de temperatura do valor do processo.
5. Calcule o valor de desvio com relação à flutuação. Por exemplo, se a temperatura estabiliza em torno de 22 °C (72 °F) com uma flutuação de 21,7 a 22,5 °C (71 a 72,5 °F), o valor de 'ATParams.Deviation' é:

$$\text{Para } ^\circ\text{C: } \frac{22,5 \text{ a } 21,7}{2} = 0,4 \quad \text{Para } ^\circ\text{F: } \frac{72,5 \text{ a } 71}{2} = 0,75$$

6. Defina o valor de desvio, caso ainda não tenha sido definido.
7. Mude a entrada “Inicialize” para “FALSA”.
8. Aguarde até que ‘AT_Warning’ exiba 2. O processo de autotune foi bem-sucedido.
9. Obtenha o valor em sintonia do “OutGains”.

Como o Autotune funciona

O processo de ajuste automático começa quando ‘Inicialize’ é configurado como FALSO (Etapa 7.) Nesse momento, a saída de controle aumenta na quantidade da ‘Etapa’ e o processo aguarda que o valor do processo alcance ou exceda o ‘primeiro pico’.

O primeiro pico é definido como:

Para a operação direta: Primeiro pico = PV1 – (12 x desvio)

Para a operação reversa: Primeiro pico = PV1 + (12 x desvio)

Onde PV1 é o valor de processo quando Inicialize está definido como FALSE.

Uma vez que o valor de processo atinge o primeiro pico, a saída de controle reduz o equivalente à Step e aguarda até que o valor de processo caia para o segundo pico.

O segundo pico é definido como:

Para a operação direta: Segundo pico = PV1 – (3 x desvio)

Para a operação reversa: Segundo pico = PV1 + (3 x desvio)

Uma vez que o valor de processo alcança ou fica abaixo do segundo pico, os cálculos começam e será gerado um conjunto de ganhos pelo parâmetro OutGains.

Localização de falhas no processo autotune

É possível dizer o que ocorre no segundo plano do processo de autotune pelas sequências da saída de controle. Aqui há algumas sequências conhecidas de saída de controle e o que significa se o autotune falhar. Para facilidade de ilustração da sequência de saída de controle, definimos:

- Carga: 50
- Etapa: 20

Sequência de saída 1: 50 → 70 → 30

Condição da sequência	Resultado do Autotune	Ação para a falha do Autotune
O valor de processo atingiu o ‘primeiro pico’ e o ‘segundo’ pico no tempo certo	Provavelmente correto	NA

Sequência de saída 2: 50 → 70 → 50

Condição da sequência	Resultado do Autotune	Ação para a falha do Autotune
O valor de processo não conseguiu atingir o 'primeiro pico'	Provavelmente não terá sucesso	Reduzir desvio ou aumentar a etapa

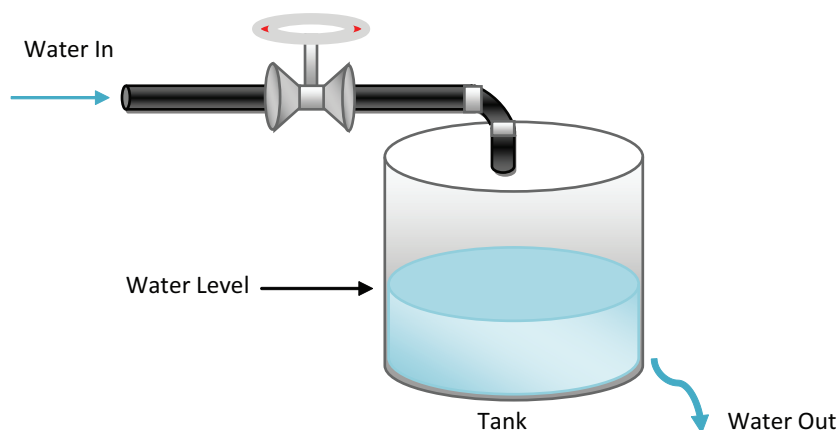
Sequência de saída 3: 50 → 70 → 30 → 50

Condição da sequência	Resultado do Autotune	Ação para a falha do Autotune
O valor de processo não conseguiu atingir o 'segundo pico'	Provavelmente não terá sucesso	Aumentar desvio ou aumentar a etapa

Sequência de saída 4: 50 → 70

Condição da sequência	Resultado do Autotune	Ação para a falha do Autotune
O valor de processo não conseguiu atingir o primeiro pico a tempo	Provavelmente não terá sucesso	Aumentar ATDynamSet

Exemplo de aplicação PID



A ilustração acima mostra um sistema de controle de nível de água básico, para manter um nível de água pré-selecionado no tanque. Uma válvula solenoide é usada para controlar a água que entra, enchendo o tanque a uma taxa pré-definida. De modo similar, a água que sai é controlada a uma taxa mensurável.

Autotune IPID para sistemas de primeira e segunda ordem

O autotune de IPID somente pode funcionar em sistemas de primeira e segunda ordem.

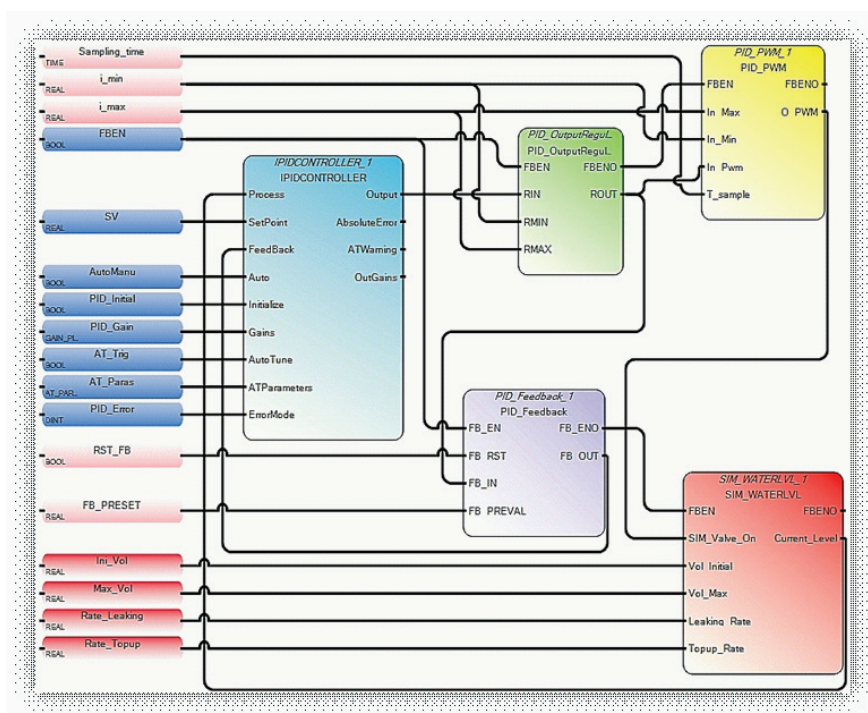
Um sistema de primeira ordem pode ser descrito por um elemento de armazenamento de energia independente simples. Exemplos de sistemas de primeira ordem são o resfriamento de um tanque de fluido, o fluxo de fluido de um tanque, um motor com torque constante conduzindo um volante a disco ou um uma rede condutora RC elétrica. O elemento de armazenamento de energia para esses sistemas são energia térmica, energia potencial, energia cinética rotacional e energia de armazenamento capacitivo, respectivamente.

Isso pode ser escrito de forma padrão como $f(t) = \tau dy/dt + y(t)$, onde τ é a constante de tempo do sistema, f é a função de força e y é a variável de estado do sistema.

No exemplo de resfriamento do tanque de fluido, ele pode ser modelado pela capacitância térmica C do fluido e resistência térmica R das paredes do tanque. A constante de tempo do sistema será RC , a função de força será a temperatura ambiente e a variável de estado do sistema será a temperatura do fluido.

Um sistema de segunda ordem pode ser descrito por dois elementos de armazenamento de energia independentes que intercambiam energia armazenada. Exemplos de sistemas de segunda ordem são um motor comandando um volante a disco com o motor acoplado ao volante por meio de um eixo com rigidez de torção ou um circuito elétrico composto de uma fonte de corrente comandando uma série LR (indutor e resistor) com um shunt C (capacitor). Os elementos de armazenamento de energia para esses sistemas são a energia cinética rotacional e a energia de mola de torção para o primeiro e a energia de armazenamento capacitiva para o último. Sistemas de acionamento de motor e sistemas de aquecimento podem ser modelados tipicamente pelo circuito elétrico C e LR .

Exemplo de código PID



A ilustração de exemplo de código PID mostra um exemplo de código para o controle do exemplo de aplicação PID mostrado anteriormente. Desenvolvido usando os diagramas de blocos de funções, ele consiste em um bloco de funções predefinido, IPIDCONTROLLER, e quatro blocos de funções definidos pelo usuário. Esses quatro são:

- **PID_OutputRegulator**
Este bloco de funções definido pelo usuário regula a saída do IPIDCONTROLLER dentro de uma faixa segura para garantir que não haja dano ao hardware utilizado no processo.

SE $R_{MIN} \leq R_{IN} \leq R_{MAX}$, então $R_{OUT} = R_{IN}$,
 SE $R_{IN} < R_{MIN}$, então $R_{OUT} = R_{MIN}$,
 SE $R_{IN} > R_{MAX}$, então $R_{OUT} = R_{MAX}$.

- **PID_Feedback**
 Este bloco de funções definido pelo usuário age como um multiplexador.

SE “FB_RST” for falso, $FB_OUT = FB_IN$;
 Se “FB_RST” for verdadeiro, então $FB_OUT = FB_PREVAL$.

- **PID_PWM**
 Este bloco de funções definido pelo usuário fornece uma função PWM, convertendo um valor real em uma saída ON/OFF relacionada ao tempo.
- **SIM_WATERLVL**
 Este bloco de funções definido pelo usuário simula o processo ilustrado no exemplo de aplicação mostrado anteriormente.

IMPORTANTE O tempo de varredura do programa do usuário é importante
 O método de autotune precisa fazer com que a saída da malha de controle oscile. Para identificar o período de oscilação, o IPID deve ser chamado com frequência suficiente para ser capaz de obter uma amostra de oscilação adequada. O tempo de varredura do programa do usuário deve ser menos da metade do período de oscilação. Na essência, o teorema de Shannon, ou Nyquist-Shannon, ou o teorema de amostragem devem ser respeitados.
 Além disso, é importante que o bloco de funções seja executado a um intervalo de tempo relativamente constante. É possível obter isso tipicamente usando interrupção STI.

Carregamento do sistema

Tabela 74 - Consumo de energia do Micro830, Micro850 e Micro870

Controlador/Módulo	Consumo de energia
Micro830, Micro850 e Micro870 (sem plug-in/expansão de E/S)	5 W 8 W 11 W
Módulos de encaixe, cada	1,44 W
Expansão de E/S (consumo de energia do barramento do sistema)	2085-IQ16 - 0,85 W 2085-IQ32T - 0,95 W 2085-IA8 - 0,75 W 2085-IM8 - 0,75 W 2085-OA8 - 0,90 W 2085-OB16 - 1,00 W 2085-0V16 - 1,00 W 2085-0W8 - 1,80 W 2085-0W16 - 3,20 W 2085-IF4 - 1,70 W 2085-IF8 - 1,75 W 2085-0F4 - 3,70 W 2085-IRT4 - 2,00 W

Calcule a energia total para seu controlador Micro830/Micro850/Micro870

Para calcular a energia total para seu controlador Micro830 e Micro850, use a seguinte fórmula:

$$\text{Energia total} = \text{Alimentação da unidade principal} + \text{Nº de encaixes} * \text{Alimentação de encaixe} + \text{Soma da alimentação de Expansão de E/S}$$

Exemplo 1:

Derive a energia total para um controlador Micro830 de 24 pontos com dois encaixes.

$$\text{Potência total} = 8 \text{ W} + 1,44 \text{ W} * 2 + 0 = \mathbf{10,88 \text{ W}}$$

Exemplo 2:

Derive a energia total para um controlador Micro850 de 48 pontos, com 3 encaixes e os módulos de Expansão de E/S 2085-IQ16 e 2085-IF4 conectados.

$$\text{Potência total} = 11 \text{ W} + 3 * 1,44 \text{ W} + 0,85 \text{ W} + 1,7 \text{ W} = \mathbf{17,87 \text{ W}}$$

Calcule a carga da fonte de alimentação CA externa para seu controlador Micro830

Para calcular a carga da fonte de alimentação CA externa:

- Obtenha a carga de corrente do sensor total. Para este exemplo, presuma que é 250 mA.
- Calcule a carga de alimentação total por sensor usando esta fórmula:
(24 V * 250 mA) 6 W.
- Derive a carga da fonte de alimentação CA externa usando esta fórmula:
Carga da fonte de alimentação CA = Energia total calculada para um sistema Micro800 com encaixe + Carga de energia total por sensor.

Como exemplo, um controlador Micro850 de 48 pontos com 2 encaixes, e Expansão de E/S 2085-IQ16 e 2085-IF4 e Corrente de sensor de 250 mA (alimentação do sensor de 6 W) terá a seguinte carga total para a fonte de alimentação CA:

Carga total para a fonte de alimentação CA = 17,87 W + 6 W = 23,87 W



ATENÇÃO: A carga máxima para a fonte de alimentação CA é limitada a 38,4 W com temperatura ambiente máxima circundante limitada a 65 °C.

Conexão a redes usando DF1

IMPORTANTE Este apêndice aplica-se somente aos controladores Micro850 (2080-L50E) e Micro870 (2080-L70E).

Tópico	Página
Protocolo DF1 full-duplex	335
Protocolo DF1 half-duplex	336
Operação da linha de controle do modem	338
Configurar parâmetros DF1 half-duplex	340
Configurar uma estação DF1 half-duplex mestre de modo padrão	341
Configure um modo DF1 half-duplex baseado em mensagem Estação mestre	345
Configurar uma estação escrava	348
Configurar o tempo limite de coleta	349
Configurar uma estação de modem de rádio	350

Os protocolos a seguir são suportados na porta serial incorporada, incluindo qualquer módulo plug-in 2080-SERIALISOL instalado, nos controladores Micro850 (2080-L50E) e Micro870 (2080-L70E) mais recentes.

- DF1 full-duplex
- DF1 half-duplex mestre/escravo
- Modem de rádio DF1

Protocolo DF1 full-duplex

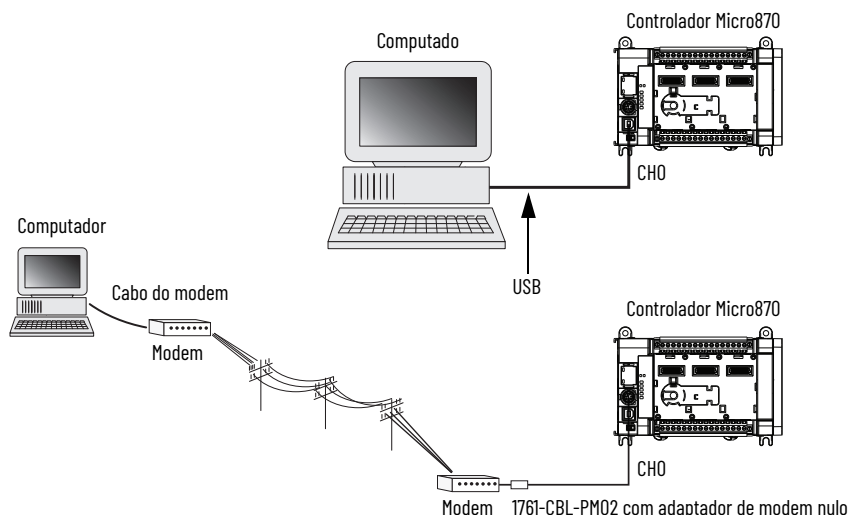
O protocolo DF1 full-duplex fornece uma conexão ponto a ponto entre dois dispositivos. O protocolo DF1 full-duplex combina transparência de dados (especificação do American National Standards Institute ANSI – X3.28-1976, subcategoria D1) com transmissão simultânea de duas vias com respostas integradas (subcategoria F1).

O controlador suporta o protocolo DF1 full-duplex via conexão RS-232 a dispositivos externos, como computadores ou outros controladores compatíveis com DF1 full-duplex.

O suporte DF1 é obtido por meio da interface serial CIP nos controladores Micro800.

O protocolo DF1 full-duplex (também chamado de protocolo ponto a ponto) DF1 é útil quando a comunicação ponto a ponto RS-232 é necessária. O protocolo DF1 controla o fluxo de mensagens, detecta e sinaliza erros e tenta novamente se forem detectados erros.

Exemplo de conexões DF1 full-duplex



Protocolo DF1 half-duplex

O protocolo DF1 half-duplex é uma rede multiponto com mestre único/múltiplos escravos. O protocolo DF1 half-duplex suporta a transparência de dados (especificação do American National Standards Institute ANSI – X3.28-1976, subcategoria D1). Diferente do DF1 full-duplex, a comunicação ocorre em uma direção de cada vez. Você pode usar a porta RS-232/RS-485 no controlador tanto como uma porta de programação half-duplex quanto uma porta de transmissão de mensagem peer-to-peer half-duplex.

Operação DF1 half-duplex

Um dispositivo DF1 half-duplex mestre inicia toda a comunicação “coletando dados” em cada dispositivo escravo. O dispositivo escravo somente pode transmitir quando for consultado pelo mestre. É responsabilidade do mestre consultar cada escravo de forma regular e sequencial para permitir aos dispositivos escravos uma oportunidade de comunicação.

Um recurso adicional do protocolo DF1 half-duplex é que é possível para um dispositivo escravo habilitar uma gravação MSG ou ler para/de outro escravo. Quando o escravo iniciador é consultado, a MSG é enviada para o mestre. O mestre reconhece que a mensagem não é destinada a ele, mas a outro escravo, então o mestre imediatamente envia a mensagem para o escravo pretendido. O mestre faz isso automaticamente; você não precisa programar o mestre para movimentar os dados entre os nós escravos. A transferência escravo-a-escravo também pode ser usada pelo software de programação para permitir o upload e o download de programas escravo-a-escravo para controladores (incluindo o mestre) no link DF1 half-duplex.

O controlador pode agir como o mestre ou como um escravo em uma rede half-duplex. Quando o controlador é um dispositivo escravo, é necessário um dispositivo mestre para executar a rede. Vários outros produtos Allen-Bradley suportam o protocolo DF1 half-duplex mestre.

O DF1 half-duplex suporta até 255 dispositivos (endereço 0 a 254) com o endereço 255 reservado para transmissões do mestre. Como um dispositivo DF1 half-duplex escravo, o controlador suporta recepção de transmissão. Como um DF1 half-duplex mestre, o controlador suporta receber e a iniciar comandos de gravação de transmissão (via instrução MSG). O controlador também suporta modems half-duplex usando reconhecimento de hardware RTS/CTS.

IMPORTANTE A versão do firmware 20 para controladores Micro850 e Micro870 não suporta a função de transmissão no mestre half-duplex. Este recurso será suportado em uma futura versão do firmware.

Considerações ao se comunicar como um DF1 escravo em um link multiponto

Quando a comunicação é entre seu software de programação e um controlador Micro800 ou entre dois controladores Micro800 por meio de comunicação escravo-a-escravo em um link multiponto maior, os dispositivos dependem de um DF1 half-duplex Master para dar a cada um deles acesso em tempo hábil. À medida que o número de dispositivos escravos aumenta, também aumenta o tempo entre o momento em que os dispositivos escravos são consultados. Este aumento no tempo também pode ser grande se você estiver usando baud rates baixas. À medida que os períodos de tempo aumentam, pode ser necessário aumentar os valores de tempo limite de coleta e de tempo limite de resposta para dispositivos escravos.

IMPORTANTE O download do programa não é suportado no DF1 half-duplex e no Modem de rádio para controladores Micro850 (2080-L50E) e Micro870 (2080-L70E) com versão do firmware 20 ou posterior.

Uso de modems com controladores programáveis Micro800

Os tipos de modems que você pode usar com os controladores Micro800 incluem os seguintes:

- **Modems de telefone dial-up.**
Um controlador Micro800, na extremidade receptora da conexão dial-up, pode ser configurado para protocolo DF1 full-duplex com ou sem reconhecimento. O modem conectado ao controlador Micro800 deve suportar a resposta automática.
- **Modems de linha alugada.**
Os modems de linha alugada são usados com linhas de telefone dedicadas, que normalmente são alugadas da empresa de telefonia local. As linhas dedicadas podem estar em uma topologia ponto a ponto compatível com comunicações full-duplex entre dois modems ou em uma topologia multiponto compatível com comunicações half-duplex entre três ou mais modems.
- **Modems de rádio.**
Os modems de rádio podem ser implementados em uma topologia ponto a ponto compatível com comunicações half-duplex ou full-duplex ou em uma topologia multiponto compatível com comunicações half-duplex entre três ou mais modems. Os controladores Micro800 também suportam protocolo de modem de rádio DF1.
- **Drivers de linha.**
Os drivers de linha, também chamados de modems de curta distância, não modulam efetivamente os dados seriais, mas condicionam os sinais elétricos a operar de forma confiável em longas distâncias de transmissão (até vários quilômetros). Os drivers de linha estão disponíveis nos modelos full-duplex e half-duplex.

Para conexões de modem full-duplex ponto a ponto que não exigem sinais de reconhecimento de modem para operar, use o protocolo DF1 full-duplex sem reconhecimento. Para conexões de modem full-duplex ponto a ponto que exigem sinais de reconhecimento RTS/CTS, use o protocolo DF1 full-duplex com reconhecimento.

Para conexões de modem de rádio, use o protocolo de modem de rádio DF1, especialmente se for necessária a capacidade de armazenar e encaminhar.

Para conexões gerais de modem multiponto ou para conexões de modem ponto a ponto que exijam reconhecimento RTS/CTS, use o protocolo DF1 half-duplex escravo. Neste caso, um (e somente um) dos outros dispositivos deve ser configurado para o protocolo DF1 half-duplex mestre.



Os controladores Micro850 (2080-L50E) e Micro870 (2080-L70E) suportam reconhecimento de modem RTS/CTS quando configurados para protocolo DF1 full-duplex, com o parâmetro linha de controle definido como reconhecimento para modem full-duplex ou protocolo DF1 half-duplex escravo com o parâmetro linha de controle definido para modem half-duplex.

Esses controladores também suportam a linha DCD (Data Carrier Detect) para o protocolo Modem de rádio DF1. Nenhuma outra linha de reconhecimento de modem (como Data-Set™ Ready e Data Terminal Ready) é suportada pelos controladores Micro800.

Operação da linha de controle do modem

A seguir é explicada a operação dos controladores Micro800 ao configurar a porta serial RS-232 para as seguintes aplicações.

DF1 full-duplex

Quando configurado para DF1 full-duplex, é executada a seguinte operação da linha de controle:

Nenhum reconhecimento selecionado

O RTS está sempre inativo (baixo). Recepções e transmissões ocorrem independentemente do estado da entrada CTS. Faça esta seleção somente quando os controladores Micro800 estiverem diretamente conectados a outro dispositivo que não exige sinais de reconhecimento.

Full-duplex (RTS sempre ligado) selecionado

O RTS está sempre ativo (alto).

As transmissões exigem que o CTS esteja ativo.

DF1 half-duplex escravo

Quando configurado para DF1 half-duplex escravo, é executada a seguinte operação da linha de controle:

Nenhum reconhecimento selecionado

O RTS está sempre inativo. Recepções e transmissões ocorrem independentemente do estado da entrada CTS. Faça esta seleção somente quando o controlador estiver conectado diretamente a outro dispositivo que não exige sinais de reconhecimento.

Half-duplex sem portadora contínua (RTS/CTS) selecionado

O RTS é ativado somente durante as transmissões (e quaisquer atrasos programados antes ou depois das transmissões). As transmissões exigem que o CTS esteja ativo.

DF1 half-duplex mestre

Ao configurar para DF1 half-duplex mestre, é executada a seguinte operação da linha de controle:

Nenhum reconhecimento selecionado

O RTS está sempre inativo. Recepções e transmissões ocorrem independentemente do estado da entrada CTS. Faça esta seleção somente quando o controlador estiver conectado diretamente a outro dispositivo que não exige sinais de reconhecimento.

Modem Full-Fuplex (RTS sempre ligado) selecionado

O RTS está sempre ativo (alto).

As transmissões exigem que o CTS esteja ativo.

Half-duplex sem portadora contínua (RTS/CTS) selecionado

O RTS fica ativo somente durante as transmissões (e quaisquer atrasos programados antes e depois das transmissões).

As transmissões exigem que o CTS esteja ativo

Modem de rádio DF1

Ao configurar os controladores Micro800 para o modem de rádio DF1, é executada a seguinte operação da linha de controle:

Nenhum reconhecimento selecionado

O RTS está sempre inativo. Recepções e transmissões ocorrem independentemente do estado da entrada CTS. Esta seleção deve ser feita somente quando o controlador estiver diretamente conectado a outro dispositivo que não exige sinais de reconhecimento.

Half-duplex sem portadora contínua (RTS/CTS) selecionado

O RTS é ativado durante a transmissão e durante quaisquer atrasos programados antes ou depois das transmissões. Os atrasos programados incluem RTS Send Delay e RTS Off Delay.

As transmissões exigem que o CTS esteja ativo. Se o CTS estiver inativo no início da transmissão, será fornecido um segundo para aguardar até que o CTS se torne ativo antes que o pacote de mensagem seja descartado.

Half-duplex com reconhecimento DCD selecionado

O RTS é ativado durante as transmissões e durante quaisquer atrasos programados antes e depois das transmissões. Os atrasos programados incluem RTS Send Delay e RTS Off Delay. O sinal de entrada DCD é monitorado para determinar se as transmissões são aceitáveis. Se o DCD estiver ativo, as recepções serão possíveis.

As transmissões exigem que o CTS esteja ativo e o DCD esteja inativo. Se o DCD estiver ativo no início da transmissão, um atraso configurado (DCD Wait Delay) irá esperar até que o DCD se torne inativo antes de descartar o pacote. Se o CTS estiver inativo no início da transmissão, será fornecido um segundo para aguardar até que o CTS se torne ativo antes que o pacote de mensagem seja descartado.

Configurar parâmetros DFI half-duplex

RTS Send Delay e RTS Off Delay

Por meio de seu software de programação, os parâmetros RTS Send Delay e RTS Off Delay permitem que você defina por quanto tempo o RTS ficará ativado antes da transmissão, bem como por quanto tempo será mantido ativado após a conclusão da transmissão. Esses parâmetros se aplicam somente quando você seleciona o modem half-duplex. Para produtividade máxima de comunicação, deixe estes parâmetros com zero.

Para uso com modems half-duplex que exigem tempo extra de resposta ou de ativação de seu transmissor. Mesmo depois de ativar o CTS, o RTS Send Delay específica (em incrementos de 20 milissegundos) o tempo de atraso após a ativação do RTS que deve ser aguardado antes de verificar se o CTS foi ativado pelo modem. Se o CTS ainda não estiver ativo, o RTS permanece ativo e a transmissão ocorrerá se o CTS for ativado em até um segundo. Após um segundo, se o CTS ainda não estiver ativado, o RTS será definido como inativo e a transmissão será cancelada.

Para modems que não fornecem um sinal CTS, mas ainda precisam que o RTS seja ativado antes da transmissão, faça o jumper RTS para CTS e use o menor atraso possível sem perder uma operação confiável.

IMPORTANTE Se for selecionado um RTS Send Delay de 0, a transmissão será iniciada assim que o CTS for ativado. Se o CTS não ficar ativo dentro de um segundo após o RTS ser ativado, o RTS será definido como inativo e a transmissão será cancelada.

Certos modems irão descartar seu link de portadora quando o RTS for definido como inativo, mesmo que a transmissão ainda não tenha sido concluída. O parâmetro RTS Off Delay específica, em incrementos 20 milissegundos, que o atraso entre o envio ao modem do último caractere serial e quando o RTS é desativado. Isso dá ao modem tempo extra para transmitir o último caractere de um pacote.



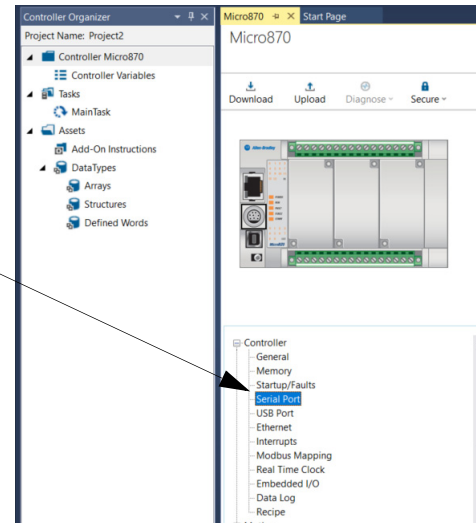
ATENÇÃO: Para quase todas as aplicações de modem, o RTS Off Delay deve ser deixado com 0. Nunca selecione um do RTS Off Delay maior que o RTS Send Delay nos outros dispositivos na rede, porque pode haver dois dispositivos tentando transmitir simultaneamente.

Configurar uma estação DF1 half-duplex mestre de modo padrão

Escolha o modo padrão se quiser consultar estações escravas para obter informações com base nos intervalos de coleta configurados pelo usuário. Este modo é usado com mais frequência em configurações gerais ponto a multiponto.

Para configurar o controlador como uma estação mestre usando a comunicação padrão, coloque o controlador no modo de programa e siga as etapas abaixo usando o software de programação:

1. Para abrir a página de configuração, clique em Serial port.



2. Na página Configuração da porta serial, selecione Half-Duplex Master para o modo DF1.
3. Escolha um modo de coleta de dados padrão.
4. Configure o restante do driver de comunicação de acordo com a [Tabela 75](#).

Controller - Serial Port

Baud Rate: 38400
Parity: None
Station Address: 1

Protocol Control

DF1 Mode: Half-Duplex Master
Media: RS232
Control Line: Half-Duplex w/o continuous carrier (RTS/CTS)
Error Detection: CRC
Polling Mode: Std, Single msg. per scan
☒ Duplicate Packet Detection

ACK Timeout: 50 x20 ms
Message Retries: 3
Pre Transmit Delay: 0 x1 ms
RTS Off Delay: 0 x20 ms
RTS Send Delay: 0 x20 ms

Polling Ranges

Priority High: 0 Normal High: 0
Priority Low: 255 Normal Low: 255
Normal Poll Group Size: 0

A tabela a seguir mostra os parâmetros para configurar um controlador Micro850 (2080-L50E) ou Micro870 (2080-L70E) como uma estação mestre usando o modo de comunicação padrão para se comunicar com estações escravas.

Tabela 75 - Configurar um controlador Micro800 como mestre usando o modo de comunicação padrão

Parâmetro	Seleções
Baud Rate	Selecione uma taxa de comunicação que seja suportada por todos os dispositivos no seu sistema. Configure todos os dispositivos no sistema para a mesma taxa de comunicação.
Paridade	A paridade fornece detecção adicional de erro em mensagem-pacote. Para implementar a verificação de paridade par, escolha Even. Para implementar a verificação sem paridade, escolha None.
Endereço do nó	Um endereço de nó identifica o controlador no link DF1 half-duplex. Cada estação em um link deve ter um endereço exclusivo. Escolha um endereço entre 0 ₁₀ e 254 ₁₀ . O endereço do nó 255 ₁₀ é o endereço de transmissão e não pode ser selecionado como o endereço individual de uma estação.

Tabela 75 - Configurar um controlador Micro800 como mestre usando o modo de comunicação padrão (Continuação)

Parâmetro	Seleções
Linha de controle	<p>Este parâmetro define o modo em que o driver opera. Escolha um método apropriado para a configuração do seu sistema:</p> <ul style="list-style-type: none"> Se você não estiver usando um modem, escolha NO HANDSHAKE. Se o modem mestre for full duplex, escolha FULL-DUPLEX (RTS ALWAYS ON). Se todos os modems no sistema forem half-duplex, escolha HALF-DUPLEX SEM PORTADORA CONTÍNUA (RTS/CTS). <p>Consulte Operação da linha de controle do modem na página 338 para uma descrição das configurações de operação da linha de controle.</p>
Deteção de erro	<p>Com esta seleção, você escolhe como o controlador verifica a precisão de cada transmissão de pacote DF1.</p> <p>BCC: esse algoritmo fornece um nível médio de segurança de dados. Ele não detecta:</p> <ul style="list-style-type: none"> transposição de bytes durante a transmissão de um pacote inserção ou exclusão de valores de dados de zero dentro de um pacote <p>CRC: esse algoritmo fornece um nível mais alto de segurança de dados.</p> <p>Selecione um método de detecção de erro que todos os dispositivos em sua configuração possam usar.</p> <p>Sempre que possível, escolha CRC.</p>
Polling Mode	<p>Se você quiser receber:</p> <ul style="list-style-type: none"> somente uma mensagem de uma estação escrava por vez, escolha STANDARD (SINGLE MESSAGE TRANSFER PER NODE SCAN). Escolha este método somente se for crítico manter o tempo de varredura da lista de coleta em um mínimo. todas mensagens que uma estação escrava tiver, escolha STANDARD (MULTIPLE MESSAGE TRANSFER PER NODE SCAN).
Detectar pacote duplicado	<p>Duplicate Detect permite que o controlador detecte se recebeu uma mensagem duplicada de sua mensagem mais recente de outra estação. Se você escolher detectar duplicado, o controlador reconhecerá (ACK) a mensagem, mas não agirá sobre ela, uma vez que já executou a tarefa da mensagem quando recebeu o comando da primeira mensagem.</p> <p>Se quiser detectar pacotes duplicados e rejeitá-los, marque este parâmetro. Se quiser aceitar pacotes duplicados e executá-los, deixe este parâmetro desmarcado.</p>
ACK Timeout	<p>O tempo, em incrementos de 20 milissegundos, que você deseja que o controlador espere por uma confirmação da mensagem enviada antes de tentar novamente a mensagem ou a gerar mensagem de erro. Este valor de tempo limite também é usado para o tempo limite de resposta de coleta. Consulte a página G-343 para obter recomendações para minimizar esse valor.</p>
RTS Off Delay	<p>Define a quantidade de tempo, em incrementos de 20 milissegundos, que decorre entre o final da transmissão da mensagem e a remoção do sinal RTS. Este atraso de tempo é um buffer para garantir que o modem transmita a mensagem, mas normalmente deve ser deixado em zero. Consulte RTS Send Delay e RTS Off Delay na página 340 para obter mais orientações sobre como ajustar este parâmetro.</p>
RTS Send Delay	<p>Define a quantidade de tempo, em incrementos de 20 milissegundos, que decorre entre a sinalização de RTS e o início da transmissão da mensagem. Esse tempo permite que o modem se prepare para transmitir a mensagem. O sinal Clear-to-Send (CTS) deve estar alto para que a transmissão ocorra. Consulte RTS Send Delay e RTS Off Delay na página 340 para obter mais orientações sobre como ajustar este parâmetro.</p>
Atraso de pré-transmissão	<p>Define a quantidade de tempo, em incrementos de 1 milissegundo, que decorre entre o momento que o controlador tem uma mensagem para enviar e quando faz a sinalização RTS.</p>
Novas tentativas de mensagem	<p>Define o número de vezes que uma estação mestre tenta novamente:</p> <ul style="list-style-type: none"> uma mensagem antes de declarar que não pode ser entregue ou um pacote de coleta de dados para uma estação ativa antes que a estação mestre declare que a estação está inativa.
Intervalo de coleta de dados prioritária - Alto	<p>Selecione o endereço da última estação escrava para a coleta de dados prioritária.</p>
Intervalo de coleta de dados prioritária - Baixo	<p>Selecione o endereço da primeira estação escrava para a coleta de dados prioritária. Inserir 255 desabilita a coleta de dados prioritária.</p>

Tabela 75 - Configurar um controlador Micro800 como mestre usando o modo de comunicação padrão (Continuação)

Parâmetro	Seleções
Intervalo de coleta de dados normal - Alto	Selecione o endereço da última estação escrava para a coleta de dados normal.
Faixa de coleta de dados normal - Baixa	Selecione o primeiro endereço da estação escrava para coleta de dados normal. Inserir 255 desabilita a coleta de dados normal.
Tamanho do grupo de coleta de dados normal	Insira a quantidade de estações ativas localizadas no intervalo de coleta de dados normal que você deseja fazer durante uma varredura no intervalo de coleta de dados normal antes de retornar à faixa de coleta de dados prioritária. Se nenhuma estação estiver configurada na Priority Polling Range, deixe este parâmetro com 0.

IMPORTANTE O valor de tempo limite não conectado na instrução MSG deve ser sempre maior que o atraso de pré-transmissão.

Tempo limite mínimo de ACK do DF1 half-duplex mestre

O parâmetro de tempo limite que determina a configuração para um DF1 half-duplex mestre é o tempo limite ACK da porta serial. O tempo limite de ACK é a quantidade de tempo que você deseja que o controlador espere por uma confirmação de suas transmissões de mensagem. Definido em intervalos de 20 milissegundos, o valor é a quantidade de tempo que o mestre irá esperar por:

- um ACK a ser retornado por um escravo quando o mestre acaba de enviar uma mensagem, ou
- uma resposta de coleta de dados ou mensagem a ser retornada por um escravo quando o mestre acaba de enviar um pacote de pesquisa.

O tempo limite deve ser longo o suficiente para que, após o mestre ter transmitido o último caractere do pacote de pesquisa, haja tempo suficiente para que um escravo transmita (e o mestre receba) um pacote de tamanho máximo antes que o tempo expire.

Para calcular o tempo limite mínimo de ACK, você precisa saber:

- a baud rate do modem
- pacote de dados de tamanho máximo (o número máximo de palavras de dados que um comando de gravação de escravo ou pacote de resposta de leitura pode conter)
- o RTS/CTS ou atraso de resposta do modem escravo
- o atraso de envio RTS configurado no escravo
- o tempo de varredura do programa do escravo

Determinação do tempo limite mínimo de ACK do mestre

Para determinar o tempo limite mínimo de ACK, você deve primeiro calcular o tempo de transmissão multiplicando tamanho máximo do pacote de dados para o seu controlador pela taxa de modem em ms/byte. Por exemplo, assumiremos um controlador Micro800 (103 palavras de dados ou total de 224 bytes de tamanho total do pacote incluindo cabeçalho) e um modem de 9.600 bps, que transmite a aproximadamente 1 ms/byte. Portanto, o tempo de transmissão da mensagem é de 224 ms. Para taxas aproximadas de transmissão do modem, consulte a tabela a seguir.

Tabela 76 - Taxas aproximadas de transmissão do modem

Velocidade (bps)	Taxa, aprox. (ms/byte)
4800	2 ms/byte
9600	2 ms/byte
19200	0,5 ms/byte

Em seguida, você precisa determinar o tempo médio de varredura do programa escravo. No software Connected Components Workbench, clique duas vezes nas variáveis do controlador no organizador do controlador e localize o tag do sistema _SYSVA_TCYCYCTIME na guia Variable. Para este exemplo, presuma que o tempo de varredura do programa é de 20 ms. O tempo de varredura do programa varia de acordo com a aplicação.

Por fim, você deve determinar o maior entre dois valores, o tempo de atraso de envio RTS do escravo configurado ou o tempo de resposta do modem escravo. O tempo de atraso de envio RTS pode ser encontrado na tela Configuration da porta serial incorporada do Micro800. Observe que o tempo de atraso de envio RTS está em intervalos de 20 ms, portanto, com um valor de 3 na caixa, o tempo de atraso de envio RTS seria de 20 ms multiplicado por 3. Usando este valor (60 ms) para o nosso exemplo, e presumindo que o tempo de resposta do modem seja de 50 ms (que varia de acordo com o modem), você deve escolher usar o tempo de atraso de envio RTS de 60 ms para o seu cálculo.

Tendo determinado o tempo máximo de transmissão de mensagem (224 ms), o tempo médio de varredura do programa escravo (20 ms) e o maior atraso de envio RTS (60 ms) ou o tempo de resposta do modem, o tempo limite mínimo de ACK é simplesmente a soma desses valores.

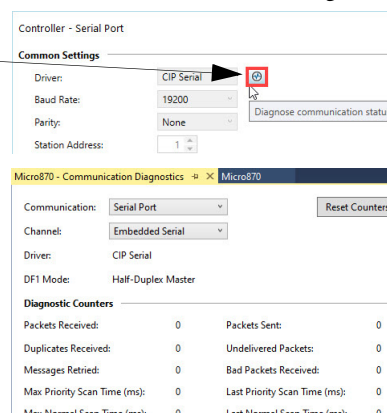
Parâmetro	Valores de exemplo (em ms)
Tempo máximo de transmissão de mensagem	224
Tempo médio de varredura do programa	20
Atraso de envio de amostragem em tempo real	60
Tempo de resposta do modem	50
Tempo limite de ACK calculado	304
Arredonde para cima para os 20 ms mais próximos	320

} Use apenas o maior desses dois valores.

Diagnóstico de comunicação do DF1 half-duplex mestre

O diagnóstico de comunicação está disponível enquanto conectado ao controlador clicando no botão Diagnose communication status. [Tabela 77](#) explica informações sobre os dados exibidos no contador de diagnóstico.

1. Clique em Diagnose communication status para exibir o diagnóstico do DF1 half-duplex mestre.



2. Consulte a [Tabela 77](#) para obter detalhes sobre a tela de diagnóstico na comunicação do DF1 half-duplex mestre.

Tabela 77 - Parâmetros de diagnóstico de comunicação do DF1 half-duplex mestre

Campo Status	Definição
Pacotes enviados	O número total de mensagens DF1 enviadas pelo controlador (incluindo novas tentativas de mensagem).
Pacotes recebidos	O número de mensagens recebidas sem erros.
Último Normal Scan Time (ms)	Tempo em incrementos de milissegundos da última varredura através da lista de coleta de dados normal.
Tempo de varredura (ms) da última prioridade	Tempo em incrementos de milissegundos da última varredura através da lista de coleta de dados prioritária.
Nova tentativa de mensagem	O número de novas tentativas de mensagem enviadas pelo controlador.
Pacotes não entregues	O número de mensagens que foram enviadas pelo controlador, mas não reconhecidas pelo dispositivo de destino.
Duplicados recebidos	O número de vezes que o controlador recebeu um pacote de mensagem idêntico ao pacote de mensagem anterior.
Pacote ruim recebido	O número de pacotes de dados incorretos recebidos pelo controlador para os quais nenhum ACK foi retornado.
Max Normal Scan Time (ms)	Tempo máximo em incrementos de milissegundos para varrer a lista de coleta de dados normal.
Max Priority Scan Time (ms)	Tempo máximo em incrementos de milissegundos para varrer a lista de coleta de dados prioritária.

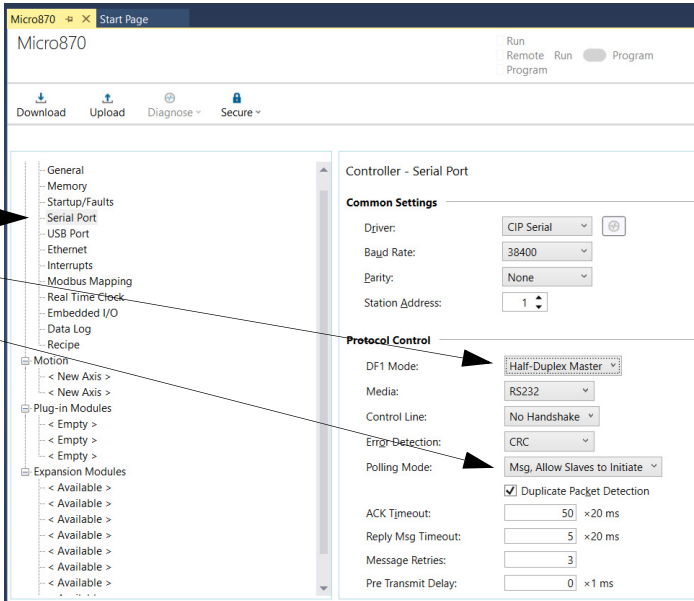
Configure um modo DF1 half-duplex baseado em mensagem Estação mestre

Escolha o modo de comunicação baseado em mensagem se quiser usar instruções MSG na programação do usuário para se comunicar com uma estação de cada vez. Se a sua aplicação usa transmissão por satélite ou transmissão por celular, considere escolher baseado em mensagem. A comunicação com uma estação escrava pode ser iniciada conforme necessário.

Com o modo baseado em mensagem, você não tem um arquivo de nó ativo que pode ser usado para monitorar o status da estação. Além disso, não é possível implementar mensagens de estação escrava a estação escrava ou programação escrava.

Para configurar o controlador para uma estação mestre usando a comunicação baseada em mensagem, coloque o controlador no modo de programa e siga as etapas abaixo no software Connected Components Workbench.

- 1. Para abrir a página de configuração, clique em Serial Port.
- 2. Na página Configuração da porta serial, selecione half-duplex mestre para o modo DF1.
- 3. Escolha um modo de coleta de dados baseado em mensagem.
- 4. Configure o restante do driver de comunicação de acordo com a [Tabela 78](#).



Defina os parâmetros mostrados na [Tabela 78](#) ao configurar um controlador Micro800 como uma estação mestre usando o modo de comunicação baseado em mensagem para se comunicar com estações escravas.

Tabela 78 - Configurar um controlador Micro800 como mestre usando o modo de comunicação baseado em mensagem

Parâmetro	Seleções
Baud Rate	Selecione uma taxa de comunicação que seja suportada por todos os dispositivos no seu sistema. Configure todos os dispositivos no sistema para a mesma taxa de comunicação.
Paridade	A paridade fornece detecção adicional de erro em mensagem-pacote. Para implementar a verificação de paridade par, escolha Even. Para implementar a verificação sem paridade, escolha None.
Endereço do nó	Um endereço de nó identifica o controlador no link DF1 half-duplex. Cada estação em um link deve ter um endereço exclusivo. Escolha um endereço entre 0 ₁₀ e 254 ₁₀ . O endereço do nó 255 ₁₀ é o endereço de transmissão e não pode ser selecionado como o endereço individual de uma estação.
Media	Selecione a mídia de comunicação para o protocolo DF1: <ul style="list-style-type: none">• RS-232• RS-485 (disponível somente quando o modo DF1 é half-duplex)
Linha de controle	Este parâmetro define o modo em que o driver opera. Escolha um método apropriado para a configuração do seu sistema: <ul style="list-style-type: none">• Se você não estiver usando um modem, escolha NO HANDSHAKE.• Se o modem mestre for full-duplex, escolha FULL-DUPLEX (RTS ALWAYS ON).• Se todos os modems no sistema forem half-duplex, escolha HALF-DUPLEX WITHOUT CONTINUOUS CARRIER (RTS/CTS). Consulte Operação da linha de controle do modem na página 338 para descrições das configurações de operação da linha de controle.
Deteção de erro	Com esta seleção, você escolhe como o controlador verifica a precisão de cada transmissão de pacote DF1. BCC: esse algoritmo fornece um nível médio de segurança de dados. Ele não detecta: <ul style="list-style-type: none">- transposição de bytes durante a transmissão de um pacote- inserção ou exclusão de valores de dados de zero dentro de um pacote CRC: esse algoritmo fornece um nível mais alto de segurança de dados. Selecione um método de detecção de erro que todos os dispositivos em sua configuração possam usar. Sempre que possível, escolha CRC.

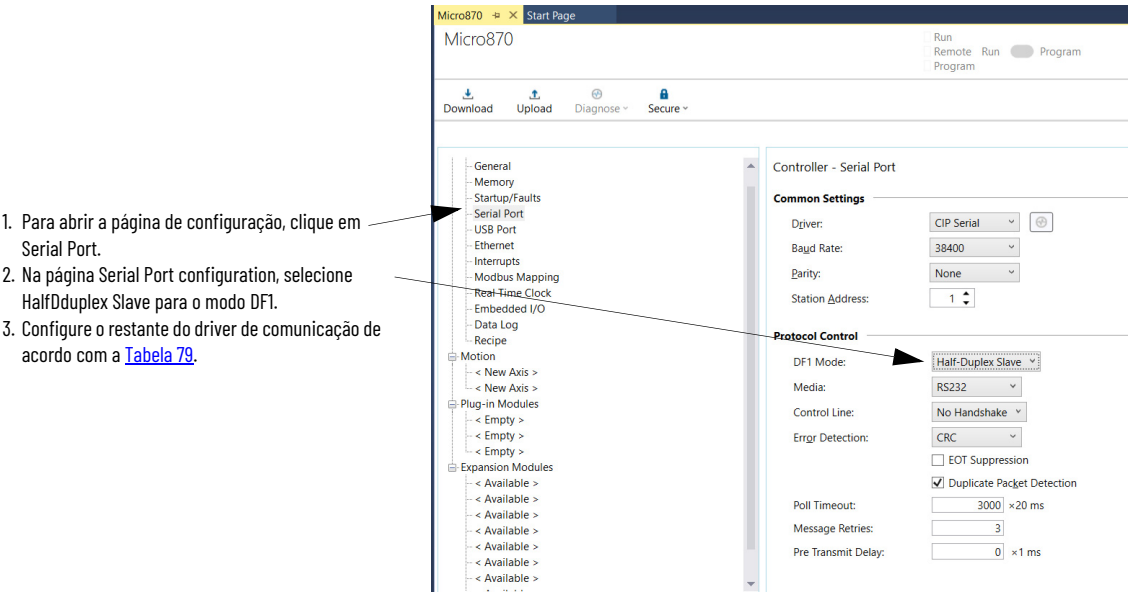
Tabela 78 - Configurar um controlador Micro800 como mestre usando o modo de comunicação baseado em mensagem (Continuação)

Parâmetro	Seleções
Polling Mode	<p>Se você deseja:</p> <ul style="list-style-type: none"> aceitar mensagens não solicitadas de estações escravas, escolha MESSAGE BASED (ALLOW SLAVES TO INITIATE MESSAGES) As mensagens iniciadas por estação escrava são reconhecidas e processadas após todas as mensagens iniciadas por estação mestre (solicitadas). Observação: as estações escravas somente podem enviar mensagens quando forem consultadas. Se a estação mestre baseada em mensagem nunca enviar uma mensagem para uma estação escrava, a estação mestre nunca enviará uma pesquisa para a estação escrava. Portanto, para obter regularmente uma mensagem iniciada pela estação escrava de uma estação escrava, você deve escolher usar o modo de comunicação padrão. ignorar mensagens não solicitadas de estações escravas, escolha MESSAGE BASED (DO NOT ALLOW SLAVES TO INITIATE MESSAGES) As mensagens iniciadas pela estação escrava são reconhecidas e descartadas. A estação mestre reconhece a mensagem iniciada pela estação escrava para que a estação escrava remova a mensagem de sua fila de transmissão, o que permite que o próximo pacote programado seja transmitido para a fila de transmissão.
Detectar pacote duplicado	<p>Duplicate Detect permite que o controlador detecte se recebeu uma mensagem duplicada de sua mensagem mais recente de outra estação. Se você escolher detectar duplicado, o controlador reconhecerá (ACK) a mensagem, mas não agirá sobre ela, uma vez que já executou a tarefa da mensagem quando recebeu o comando da primeira mensagem. Se quiser detectar pacotes duplicados e rejeitá-los, marque este parâmetro. Se quiser aceitar pacotes duplicados e executá-los, deixe este parâmetro desmarcado.</p>
Tempo limite de espera da mensagem de resposta	<p>Define a quantidade de tempo, em incrementos de 20 milissegundos, que a estação mestre irá esperar após receber um ACK (para uma mensagem iniciada pelo mestre) antes de fazer a coleta de uma resposta da estação escrava. Escolha um tempo que seja, no mínimo, igual ao tempo mais longo que uma estação escrava precisa para formatar um pacote de resposta. Geralmente, é o tempo máximo de varredura da estação escrava.</p>
ACK Timeout	<p>O tempo, em incrementos de 20 milissegundos, que você deseja que o controlador espere por uma confirmação da mensagem enviada antes de tentar novamente a mensagem ou a gerar mensagem de erro. Este valor de tempo limite também é usado para o tempo limite de resposta de coleta. Consulte Tempo limite mínimo de ACK do DF1 half-duplex mestre na página 343 para obter recomendações sobre como minimizar esse valor.</p>
RTS Off Delay	<p>Define a quantidade de tempo, em incrementos de 20 milissegundos, que decorre entre o final da transmissão da mensagem e a remoção do sinal RTS. Este atraso de tempo é um buffer para garantir que o modem transmita a mensagem, mas normalmente deve ser deixado em zero. Consulte RTS Send Delay e RTS Off Delay na página 340 para obter mais orientações sobre como ajustar este parâmetro.</p>
RTS Send Delay	<p>Define a quantidade de tempo, em incrementos de 20 milissegundos, que decorre entre a sinalização de RTS e o início da transmissão da mensagem. Esse tempo permite que o modem se prepare para transmitir a mensagem. O sinal Clear-to-Send (CTS) deve estar alto para que a transmissão ocorra. Consulte RTS Send Delay e RTS Off Delay na página 340 para obter mais orientações sobre como ajustar este parâmetro.</p>
Atraso de pré-transmissão	<p>Define a quantidade de tempo, em incrementos de 1 milissegundo, que decorre entre o momento em que o controlador tem uma mensagem para enviar e quando faz a sinalização de RTS.</p>
Novas tentativas de mensagem	<p>Define o número de vezes que uma estação mestre tenta novamente uma mensagem antes de declarar que ela não pode ser entregue.</p>

IMPORTANTE O valor de tempo limite não conectado na instrução MSG deve ser sempre maior que o atraso de pré-transmissão.

Configurar uma estação escrava

Para escolher o controlador como uma estação escrava, siga as etapas abaixo usando seu software de programação:



Defina estes parâmetros ao configurar um controlador Micro800 como uma estação escrava.

Tabela 79 - Configurar um controlador Micro800 como uma estação escrava

Parâmetro	Seleções
Baud Rate	Selecione uma taxa de comunicação que seja suportada por todos os dispositivos no seu sistema. Configure todos os dispositivos no sistema para a mesma taxa de comunicação.
Paridade	A paridade fornece detecção adicional de erro em mensagem-pacote. Para implementar a verificação de paridade par, escolha Even. Para implementar a verificação sem paridade, escolha None.
Endereço do nó	Um endereço de nó identifica o controlador no link DF1 half-duplex. Cada estação em um link deve ter um endereço de nó exclusivo. Escolha um endereço entre 0 ₁₀ e 254 ₁₀ . O endereço do nó 255 ₁₀ é o endereço de transmissão e não pode ser selecionado como o endereço individual de uma estação.
Linha de controle	Este parâmetro define o modo em que o driver opera. Escolha um método apropriado para a configuração do seu sistema: <ul style="list-style-type: none"> Se você não estiver usando um modem, escolha NO HANDSHAKE. Se o modem mestre for full-duplex e o modem escravo for half-duplex, escolha HALF-DUPLEX WITHOUT CONTINUOUS CARRIER (RTS/CTS). Consulte Operação da linha de controle do modem na página 338 para descrições das configurações de operação da linha de controle.
Detecção de erro	Com esta seleção, você escolhe como o controlador verifica a precisão de cada transmissão de pacote DF1. BCC: esse algoritmo fornece um nível médio de segurança de dados. Ele não detecta: <ul style="list-style-type: none"> transposição de bytes durante a transmissão de um pacote inserção ou exclusão de valores de dados de zero dentro de um pacote CRC: esse algoritmo fornece um nível mais alto de segurança de dados. Selecione um método de detecção de erro que todos os dispositivos em sua configuração possam usar. Sempre que possível, escolha CRC.
Detectar pacote duplicado	Duplicate Detect permite que o controlador detecte se recebeu uma mensagem duplicada de sua mensagem mais recente da estação mestre. Se você escolher detectar duplicado, o controlador reconhecerá (ACK) a mensagem, mas não agirá sobre ela, uma vez que já executou a tarefa da mensagem quando recebeu o comando da primeira mensagem. Se quiser detectar pacotes duplicados e rejeitá-los, marque este parâmetro. Se quiser aceitar pacotes duplicados e executá-los, deixe este parâmetro desmarcado.

Tabela 79 - Configurar um controlador Micro800 como uma estação escrava (Continuação)

Parâmetro	Seleções
Tempo limite de coleta	<p>O temporizador monitora a frequência com que a estação é consultada. Se a estação tiver uma mensagem para enviar, ela inicia um temporizador.</p> <p>Se o tempo limite de coleta expirar antes do tempo limite da mensagem especificado no bloco de controle MSG, o bit de erro MSG é definido e a mensagem é removida da fila de transmissão.</p> <p>Se o tempo limite da mensagem, especificado no bloco de controle MSG, expirar antes do tempo limite de coleta expirar, o bit de erro MSG e o bit de tempo limite MSG serão definidos.</p> <p>O tempo limite de coleta pode ser desabilitado inserindo um zero. Consulte Configurar o tempo limite de coleta na página 349 para recomendações para minimizar esse valor.</p>
RTS Off Delay	<p>Define a quantidade de tempo, em incrementos de 20 milissegundos, que decorre entre o final da transmissão da mensagem e a remoção do sinal RTS. Este atraso de tempo é um buffer para garantir que o modem transmita a mensagem, mas normalmente deve ser deixado em zero. Consulte RTS Send Delay e RTS Off Delay na página 340 para obter mais orientações sobre como ajustar este parâmetro.</p>
RTS Send Delay	<p>Define a quantidade de tempo, em incrementos de 20 milissegundos, que decorre entre a sinalização de RTS e o início da transmissão da mensagem. Esse tempo permite que o modem se prepare para transmitir a mensagem. O sinal Clear-to-Send (CTS) deve estar alto para que a transmissão ocorra. Consulte RTS Send Delay e RTS Off Delay na página 340 para obter mais orientações sobre como ajustar este parâmetro.</p>
Novas tentativas de mensagem	<p>Define o número de vezes que uma estação escrava reenvia sua mensagem para a estação mestre antes que a estação escrava declare que a mensagem não pode ser entregue.</p>
Atraso de pré-transmissão	<p>Define a quantidade de tempo, em incrementos de 1 milissegundo, que decorre entre o momento que o controlador tem uma mensagem para enviar e quando faz a sinalização del RTS.</p>
Supressão EOT	<p>Se quiser minimizar o tráfego na rede, você pode escolher que a estação escrava não envie pacotes de EOT para a estação mestre. Quando os pacotes de EOT são suprimidos, a estação mestre assume automaticamente que uma estação escrava não tem dados a fornecer se a estação escrava não enviar um pacote de mensagem como resposta a uma pesquisa.</p> <p>Uma desvantagem de suprimir EOTs é que a estação mestre não consegue distinguir entre uma estação ativa que não tem dados para transmitir e uma estação inativa. Uma possível aplicação para suprimir EOTs é a seguinte: conservar a alimentação com um modem de rádio porque o transmissor de rádio não precisa energizar para transmitir um pacote DLE EOT (pacote "sem dados para dar").</p> <p>Para suprimir EOTs, marque este parâmetro. Para que o controlador envie EOTs, deixe este parâmetro desmarcado.</p>

Configurar o tempo limite de coleta

O tempo limite de coleta é usado somente quando o DF1 half-duplex escravo está iniciando instruções MSG na lógica ladder. Isso implica que o mestre provavelmente está configurado para o modo de coleta de dados padrão. O valor mínimo de tempo limite de coleta depende da taxa máxima de varredura de coleta de dados do mestre. Como a coleta de dados do mestre e o acionamento do escravo de uma instrução MSG são eventos assíncronos, é possível que, no instante seguinte à que o escravo for consultado, seja disparada a instrução MSG. Isso significa que a instrução MSG permanecerá enfileirada para transmissão até que o mestre tenha consultado todos os demais escravos primeiro. Portanto, o valor mínimo de tempo limite de coleta da porta serial do escravo é igual à taxa máxima de varredura de coleta de dados mestre arredondada para cima até o próximo incremento de 20 ms.

Tempo limite mínimo de coleta de dados da porta serial = taxa máxima de coleta de dados da varredura mestre

Diagnóstico de comunicação do DF1 half-duplex escravo

O diagnóstico de comunicação está disponível enquanto conectado ao controlador clicando no botão Diagnose communication status. [Tabela 80](#) explica informações sobre os dados exibidos no contador de diagnóstico.

1. Clique em Diagnose communication status para exibir o diagnóstico do DF1 half-duplex escravo.

2. Consulte a [Tabela 80](#) para obter detalhes sobre a tela de diagnóstico na comunicação do DF1 half-duplex escravo.

Tabela 80 - Parâmetros de Diagnóstico de comunicação do DF1 half-duplex escravo

Campo Status	Definição
Pacotes enviados	O número total de mensagens DF1 enviadas pelo controlador (incluindo novas tentativas de mensagem).
Pacotes recebidos	O número de mensagens recebidas sem erros.
NAKs recebidos	O número de NAKs recebidos pelo controlador.
Sem memória para receber	O número de vezes que o controlador não pôde receber uma mensagem porque não tinha memória disponível.
Novas tentativas de mensagens	O número de novas tentativas de mensagem enviadas pelo controlador.
Pacotes não entregues	O número de mensagens que foram enviadas pelo controlador, mas não reconhecidas pelo dispositivo de destino.
Duplicados recebidos	O número de vezes que o controlador recebeu um pacote de mensagem idêntico ao pacote de mensagem anterior.
Pacotes ruins recebidos	O número de pacotes de dados incorretos recebidos pelo controlador para os quais nenhum ACK foi retornado.

Configurar uma estação de modem de rádio

Para configurar o canal 1 de um controlador Micro800 para o modem de rádio DF1, faça o seguinte usando o software de programação:

1. Para abrir a página de configuração, clique em Serial Port.
2. Na página Configuração da porta serial, selecione Modem de rádio para o modo DF1.
3. Configure o restante do driver de comunicação de acordo com a [Tabela 81](#).

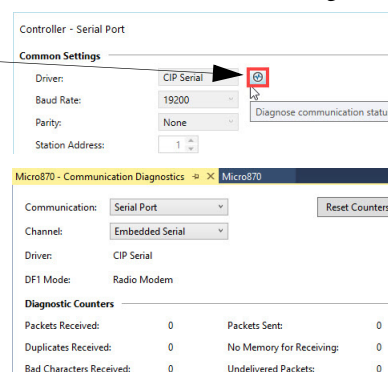
Tabela 81 - Configure um controlador Micro800 para comunicação por modem de rádio DF1

Parâmetro	Padrão	Seleções
Baud Rate	19.200	Selecione uma taxa de comunicação que seja suportada por todos os dispositivos no seu sistema. Configure todos os dispositivos no sistema para a mesma taxa de comunicação.
Paridade	Nenhuma	A paridade fornece detecção adicional de erro em mensagem-pacote. Para implementar a verificação de paridade par, escolha Even. Para implementar a verificação de paridade ímpar, escolha Odd. Para implementar a verificação sem paridade, escolha None.
Endereço do nó	1	Um endereço de nó identifica o controlador no link DF1 half-duplex. Cada estação em um link deve ter um endereço de nó exclusivo. Escolha um endereço entre 0 ₁₀ e 254 ₁₀ . O endereço do nó 255 ₁₀ é o endereço de transmissão e não pode ser selecionado como o endereço individual de uma estação.
Enable Store and Forward	Não selecionado (desabilitado)	Consulte Configurar a tabela de armazenamento e encaminhamento na página 352 para obter mais informações.
Linha de controle	No Handshake	Este parâmetro define o modo em que o driver opera. Escolha um método apropriado para a configuração do seu sistema: <ul style="list-style-type: none"> Se você não estiver usando um modem, escolha NO HANDSHAKE. Half-duplex sem portadora contínua (RTS/CTS) Half-duplex com reconhecimento DCD Consulte Operação da linha de controle do modem na página 338 para descrições das configurações de operação da linha de controle.
Deteção de erro	CRC	Com esta seleção, você escolhe como o controlador verifica a precisão de cada transmissão de pacote DF1. <p>BCC: esse algoritmo fornece um nível médio de segurança de dados. Ele não detecta:</p> <ul style="list-style-type: none"> transposição de bytes durante a transmissão de um pacote inserção ou exclusão de valores de dados de zero dentro de um pacote <p>CRC: esse algoritmo fornece um nível mais alto de segurança de dados.</p> <p>Selecione um método de detecção de erro que todos os dispositivos em sua configuração possam usar. Sempre que possível, escolha CRC.</p>
Atraso na desenergização da amostragem em tempo real	0	Define a quantidade de tempo, em incrementos de 20 milissegundos, que decorre entre o final da transmissão da mensagem e a remoção do sinal RTS. Este atraso de tempo é um buffer para garantir que o modem transmita a mensagem, mas normalmente deve ser deixado em zero. Consulte RTS Send Delay e RTS Off Delay na página 340 para obter mais orientações sobre como ajustar este parâmetro.
RTS On Delay	0	Define a quantidade de tempo, em incrementos de 20 milissegundos, que decorre entre a sinalização de RTS e o início da transmissão da mensagem. Esse tempo permite que o modem se prepare para transmitir a mensagem. O sinal Clear-to-Send (CTS) deve estar alto para que a transmissão ocorra. Consulte RTS Send Delay e RTS Off Delay na página 340 para obter mais orientações sobre como ajustar este parâmetro.
Atraso de pré-transmissão	0	Define a quantidade de tempo, em incrementos de 1 milissegundo, que decorre entre o momento que o controlador tem uma mensagem a enviar e quando ele faz a sinalização de RTS (se for selecionado reconhecimento) ou começa a transmitir (se for selecionado sem reconhecimento).

Diagnóstico de comunicação do modem de rádio DF1

O diagnóstico de comunicação está disponível enquanto conectado ao controlador clicando no botão Diagnose communication status. [Tabela 82](#) explica informações sobre os dados exibidos no contador de diagnóstico.

1. Clique em Diagnose communication status para exibir o diagnóstico do modem de rádio DF1.



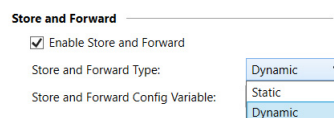
2. Consulte a [Tabela 82](#) para obter detalhes sobre a tela de diagnóstico na comunicação do Modem de rádio DF1.

Tabela 82 - Parâmetros de diagnóstico de comunicação do modem de rádio DF1

Campo Status	Definição
Pacotes enviados	O número total de mensagens DF1 enviadas pelo controlador.
Pacotes recebidos	O número de mensagens recebidas sem erros.
Sem memória para receber	O número de vezes que o controlador não pôde receber uma mensagem porque não tinha memória disponível.
Pacotes não entregues	O número de mensagens que foram enviadas pelo controlador, mas não reconhecidas pelo dispositivo de destino.
Duplicados recebidos	O número de vezes que o controlador recebeu um pacote de mensagem idêntico ao pacote de mensagem anterior.
Caracteres ruins recebidos	O número de caracteres de dados recebidos pelo controlador com erros de transmissão.

Configurar a tabela de armazenamento e encaminhamento

A função armazenar e encaminhar nos controladores Micro800 fornece dois métodos de configuração – dinâmica, que requer a criação de uma tabela armazenar e encaminhar, ou estática, que pode ser configurada diretamente no software Connected Components Workbench.



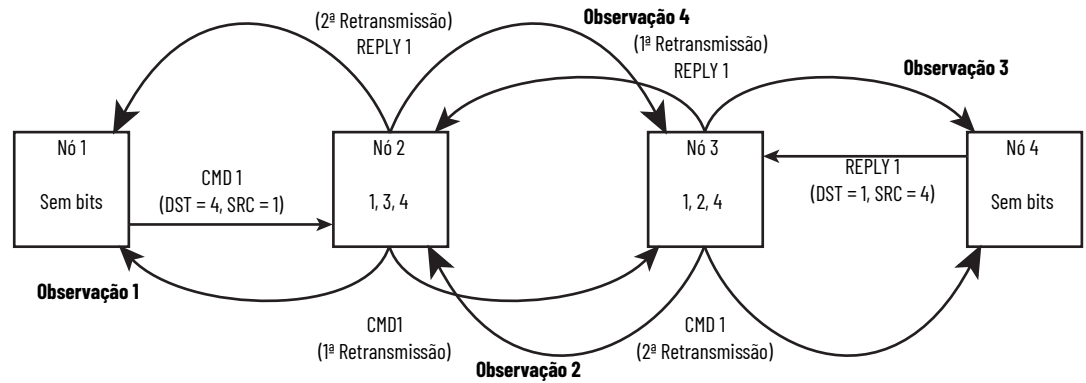
Use o método estático quando quiser configurar as tabelas Store e Forward enquanto o controlador estiver off-line. Use o método dinâmico quando quiser configurar as tabelas Store e Forward enquanto o controlador estiver no modo RUN. A tabela dinâmica Store and Forward ocupa um tipo de dados de vetor BOOL [0 a 255] ou vetor DWORD [0 a 7]. Cada bit neste vetor corresponde a um endereço de nó do modem de rádio DF1.

Para configurar um controlador Micro800 para armazenar e encaminhar pacotes de mensagens entre dois outros nós, os bits correspondentes aos endereços desses dois outros nós devem ser definidos. Por exemplo, se o nó 2 for usado para armazenar e encaminhar pacotes de mensagens entre os nós 1 e 3, então Bit 1 e Bit 3 devem ser definidos na tabela Armazenamento e encaminhamento (consulte a [Figura 30](#)).

IMPORTANTE

Uma vez que Store and Forward esteja habilitado, a detecção de pacotes duplicados também é automaticamente habilitada. Sempre que Store and Forward for usado dentro de uma rede de modem de rádio, cada nó deve ter Store and Forward habilitado, mesmo se todos os bits no arquivo forem removidos, de modo que os pacotes duplicados serão ignorados.

Figura 30 - Aplicação de protocolo de armazenamento e encaminhamento com modem de rádio DF1



Observação 1 - A camada de link do Nó 1 bloqueia a retransmissão de um pacote que é recebido com o byte SRC igual ao endereço de estação do nó receptor. Os pacotes recebidos originados do nó de recebimento nunca devem ser retransmitidos.

Observação 2 - Para impedir que o Nó 2 retransmita um pacote duplicado, a camada de link do Nó 2 atualiza a tabela de pacotes duplicados com os últimos 20 pacotes recebidos.

Observação 3 - A camada de link do Nó 4 bloqueia a retransmissão de um pacote que é recebido com o byte SRC igual ao endereço de estação do nó receptor. Os pacotes recebidos originados do nó de recebimento nunca devem ser retransmitidos.

Observação 4 - Para impedir que o Nó 3 retransmita um pacote duplicado, a camada de link do Nó 3 atualiza a tabela de pacotes duplicados com os últimos 20 pacotes recebidos.

Figura 31 - Armazenar e encaminhar tabela para nó 2, método estático

Store and Forward

☒ Enable Store and Forward

Store and Forward Type: Static

0 - 31	32 - 63	64 - 95	96 - 127	128 - 159	160 - 191	192 - 223	224 - 255
0	No Forward	8	No Forward	16	No Forward	24	No Forward
1	Forward	9	No Forward	17	No Forward	25	No Forward
2	No Forward	10	No Forward	18	No Forward	26	No Forward
3	Forward	11	No Forward	19	No Forward	27	No Forward
4	Forward	12	No Forward	20	No Forward	28	No Forward
5	No Forward	13	No Forward	21	No Forward	29	No Forward
6	No Forward	14	No Forward	22	No Forward	30	No Forward
7	No Forward	15	No Forward	23	No Forward	31	No Forward

Figura 32 - Armazenar e encaminhar tabela para nó 2, método dinâmico

Método dinâmico com vetor BOOL

Dynamic_BOOL							
Dynamic_BOOL[0]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	[0..255]
Dynamic_BOOL[1]		<input checked="" type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[2]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[3]		<input checked="" type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[4]		<input checked="" type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[5]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[6]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[7]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[8]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[9]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[10]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[11]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	
Dynamic_BOOL[12]		<input type="checkbox"/>	N/A		<input type="checkbox"/>	BOOL	

Método dinâmico com vetor DWORD

Dynamic_DWORD							
Dynamic_DWORD[0]		26	N/A		<input type="checkbox"/>	DWORD	[0..7]
Dynamic_DWORD[1]		0	N/A		<input type="checkbox"/>	DWORD	
Dynamic_DWORD[2]		0	N/A		<input type="checkbox"/>	DWORD	
Dynamic_DWORD[3]		0	N/A		<input type="checkbox"/>	DWORD	
Dynamic_DWORD[4]		0	N/A		<input type="checkbox"/>	DWORD	
Dynamic_DWORD[5]		0	N/A		<input type="checkbox"/>	DWORD	
Dynamic_DWORD[6]		0	N/A		<input type="checkbox"/>	DWORD	
Dynamic_DWORD[7]		0	N/A		<input type="checkbox"/>	DWORD	

Observações:

Symbols

__SYSVA_CYCLOCNT 138
 __SYSVA_TCYCURRENT 138
 __SYSVA_TCYMAXIMUM 138

Numerics

1761-CBL-PM02 65
 2080-PS120-240 VAC 40
 2711P-CBL-EX04 22

A

Acesso exclusivo 221
Alteração da configuração do modo de operação (RMCC) 28
 mensagem em malha reversa 29
 usando EtherNet/IP 31
 usando Modbus RTU 29
 verificar alteração do endereço do nó 30
 verificar alteração do endereço IP 32
Alterar modo de operação (RMC) 24
 alterações não confirmadas 25
 benefícios 24
 limitações 27
 memória RMC 26
 usando 291
antes de chamar a assistência 322
Aprovação norte-americana para uso em áreas classificadas 33
arquivo de função de contador de alta velocidade 211
arquivo de função de interrupção de entrada de evento (EII) 309
arquivo de função EII 309
arquivo de função HSC 211
ASCII 59, 61, 66
 configuração 71
aterramento do cabo analógico 56
aterrando o controlador 50
Atraso de envio de amostragem em tempo real 342, 347, 349
Atraso na desenergização da amostragem em tempo real 342, 347, 349
atualização do estado do eixo 159
AutoTune 328

B

Baud rate 341, 346, 348, 351
BCC 342, 346, 348, 351
Bloco de funções definido pelo usuário (UDFB) 137, 141
Bloco de funções HSC (Contador de alta velocidade) 211, 309
Bloco de funções HSC_SET_STS 213
Blocos de função PID 323

blocos de funções de controle de movimento 149

blocos de funções de movimento 145

C

cabos
 portaserial 22
 programação 21
Cabos para porta serial incorporada 22
Características gerais da chave de fim de curso programável 191
características gerais de hardware 15
Características gerais do contador de alta velocidade 191
carga do controlador 139
Cartão microSD
 atualizar flash 267
cartão microSD 238
chave de entrada do sensor de toque 146, 147
chave de fim de curso inferior (negativo) 146, 147
chave de fim de curso programável 191
chave de fim de curso superior (positivo) 146, 147
chave de posição inicial absoluta 146, 147
chave seletora 226
ciclo de varredura do programa 139
ciclo ou varredura Micro800 137
CIP Serial 66
circuito do relé de controle mestre
 testes periódicos 33
circuitos de segurança 33
Cliente DHCP 60
Cliente/servidor CIP serial 59, 61
Cliente/servidor CIP Symbolic 59, 62
Cliente/Servidor Modbus/TCP 60, 61
códigos de erro 315, 316
Comunicação
 configurando a estação de modem de rádio DF1 351
 configurando estação escrava 348
comunicações
 portas 59
Conexão DF1 ponto a ponto 65
conexões de comunicação 59, 77
ConfigMeFirst.txt
 erros 234
Configuração
 DF1 half-duplex mestre
 modo padrão 341
 Estação escrava 348
 Mínima do DF1 half-duplex mestre 343
 Tempo limite mínimo de ACK do canal 0 343
 tempo limite mínimo de ACK do mestre 344
Configuração da função EII 309
Configuração da função STI 307

Configuração de interrupção HSC 217
Configuração de interrupções do usuário 299
Configuração e status da função de interrupção de entrada de evento (EII) 309
Configuração e status da interrupção selecionada em função do tempo (STI) 307
Configuração Endian 259
Configurando
 DF1 half-duplex mestre baseado em mensagem 345
Configurar
 Tempo limite de coleta 349
configurar
 comunicação por modem de rádio DF1 351
Configuring 345
conjuntos de receita 243
Connected Components Workbench 13, 23, 137, 141, 159, 223, 224, 237, 243
considerações de alimentação
 características gerais 34
 energização da fonte de alimentação 34
 estados de entrada ao cortar a alimentação 35
 outras condições da linha 35
 perda da fonte de alimentação 34
 transformadores de isolamento 34
Considerações de segurança 32
considerações de segurança 32
 circuito do relé de controle mestre
 testes periódicos 33
 circuitos de segurança 33
 desconectando a alimentação principal 33
 distribuição de potência 33
 local perigoso 33
 testes periódicos do circuito do relé de controle mestre 33
Contador de alta velocidade (HSC) 192
controlador
 aterrando 50
 descrição 17
 fiação de E/S 55
 minimizando o ruído elétrico 55
 prevenindo aquecimento excessivo 35
Controladores Micro830
 tipos de entradas/saídas 20
controladores Micro830 16
Controladores Micro850
 tipos de entradas/saídas 20, 21
Controladores Micro870
 tipos de entradas/saídas 21
controle de geração de eventos 124
controle de movimento 143, 145
 blocos de funções administrativas 149
 entrada/saída de fiação 147
 regras gerais 150
ControlFLASH 226
CRC 342, 346, 348, 351

D

Datasets 237, 239

Deceleration 151
desconectando a alimentação principal 33
Desempenho, MSG_MODBUS 264
Deteção de erro 342, 346, 348, 351
Deteção de pacote duplicado 342, 347, 348
DF1 full-duplex
 diagnóstico de comunicação 351
diagrama de estado do eixo 158
diagrama de fiação 51
diagramas de tempo
 encoder de quadratura 200
dimensões de montagem 39
Dimensões de montagem do controlador 39
Direção PTO 146, 147
distribuição de potência 33
DLG
 lista de ID de erros de função 240
 parâmetros de entrada e saída 239
 status do bloco de funções 239
DLG_ERR_DATAFILE_ACCESS 239
DNP3
 camada de aplicação escrava 103
 diagnóstico 128
 objeto de atributo de dispositivo 119
 objetos 108
Driver de estação remota 341, 348
Driver DF1 full-duplex 351
Driver DF1 half-duplex 346, 348
Driver serial CIP
 configurar 66

E

eixo 145
Encerramento 66
encoder
 quadratura 200
encoder de quadratura 200
Endereçamento
 definição 341, 346, 348
Endereçamento CIP Symbolic 63
Endereço da estação 341, 346, 348
endereço IP
 regras 74
energização da fonte de alimentação
 considerações de alimentação 34
entrada de direção 151
entrada de posição/distância 150
entrada de velocidade 150
entradas analógicas
 orientações para fiação de canal analógico 55
entradas jerk
 regras gerais 151
erro 154
ErrorStop 158
Espaçamento do módulo 40
Espaço de endereço de mapeamento e tipos de dados suportados 259
Estabelecendo comunicações entre RSLinx e um Micro830 via USB 271

Estação full-duplex 351
Estação remota
 configuração 345
 configurando 346
 configurando o Micro800 348
 modos disponíveis 342, 346, 348
Estação remota Micro800 348
estados de eixo 159
estados de entrada ao cortar a alimentação 35
Estrutura de dados HSC APP 196
Estrutura de dados PLS 214
estrutura de dados STS HSC 205
Ethernet
 configurar as definições 72
EtherNet/IP Cliente/Servidor 60
Evento de relatório por resposta com coleta 124
evento de relatório por resposta não solicitada 125
exclusividade de saída 151, 152
Exclusões de endereço IP 74
Exemplo de aplicação PID 330
Exemplo de código PID 331
Exemplo de PLS 216
Exemplos de fiação 57

F

falhas
 recuperáveis e não recuperáveis 300
Fiação para porta serial incorporada 58
flags de qualidade do objeto 117
fonte de alimentação
 perda da 34
Forçando E/S 289
Função de chave de fim de curso programável (PLS) 214
Função definida pelo usuário (UDF) 137, 141

G

geração de eventos DNP3 121
gerenciamento de erros
 regras gerais 154

I

indicador de status 16
 comunicação serial 314
 Ethernet 22
 status da alimentação 314
 status de entrada 314
 status de falha 314
 status de operação 314
 status de rede 22, 315
 status de saída 314
 status do módulo 22, 314
Indicadores de status no controlador 313
Informação de status de interrupção HSC 219
Informações de status da função EII 310
Informações de status da função STI 307
Informações sobre o uso de interrupções 297

Início rápido 265
instalando o controlador 47
instalando o seu controlador 39
instrução de distribuição de interrupção do usuário 304
instrução de início com tempo selecionável 301
instrução de sub-rotina de interrupção 300, 301
instrução desabilitar interrupção do usuário 301
instrução habilitar interrupção do usuário 303
instrução INT 300, 301
instrução STS 301
instrução UID 301
instrução UIE 303
instrução UIF 304
interrupções
 Início selecionado em função do tempo (STS) 301
 instrução de distribuição de interrupção do usuário (UIF) 304
 instrução desabilitar interrupção do usuário (UID) 301
 instrução habilitar interrupção do usuário (UIE) 303
 instruções de interrupção 300
 rotina de falha do usuário 300
Interrupções HSC 217
interrupçõesvisão geral 297
IPIDCONTROLLER 326

L

ligando a assistência 322
limites de conexão 60
Linha de controle 342, 346, 348, 351
localização de falhas 313

M

manutenção 137, 238
Mapeamento Modbus 259
Mapeamento Modbus para Micro800 259
Marcador de posição inicial 146
MC_AbortTrigger 149
MC_Halt 150, 154, 156, 158
MC_Home 149
MC_MoveAbsolute 149, 154
MC_MoveRelative 149, 154
MC_MoveVelocity 149, 154
MC_Power 149
MC_ReadAxisError 149
MC_ReadBoolParameter 149
MC_ReadParameter 149
MC_ReadStatus 149
MC_Reset 149, 158
MC_SetPosition 149
MC_Stop 150, 154, 158
MC_TouchProbe 149

MC_WriteBoolParameter 149
MC_WriteParameter 149
Mensagem de cliente CIP 63
Micro800
 Diagnóstico de comunicação 345
minimizando o ruído elétrico 55
minimizando o ruído elétrico nos canais analógicos 56
Modbus RTU 59, 61, 66
 configuração 70
modelo de recuperação de erro 321
Modelo de recuperação de erro do controlador 321
modems
 usando com controladores Micro800 337
modo de programa 237
montagem do sistema
 controladores de 24 pontos Micro830 e Micro850 44
 controladores de 24 pontos Micro830, Micro850 e Micro870 43
montagem em painel 41
 dimensões 42
Montagem em trilho DIN 41
montagem em trilho DIN 41
movimento relativo versus movimento absoluto
 regras gerais 154

O

Operação normal 315
Operação PLS 215
Orientações e limitações para usuários avançados 141
orientações para fiação de canal analógico 55

P

Parâmetros de configuração da camada de aplicação escrava
 DNP3 88
parâmetros de configuração da camada do link 80
parâmetros de entrada 150
Parâmetros do IPIDCONTROLLER 324, 325, 326
Paridade 341, 346, 348, 351
partidas de motor (cód. cat. 509)
 supressores de transientes 49
PID 324
Ponto-a-ponto 351
porta de combinação RS-232/RS-485 59
porta Ethernet RJ-45 22, 59
porta serial
 configurar 66
porta serial RS-232/RS-485 59
POU (Unidade organizacional de programa) 138
POU da interrupção HSC 218
prevenção de colisão 126
prevenindo aquecimento excessivo 35
Prioridade de interrupções do usuário 298

proteção contra aquecimento 35
Protocolo DF1 full-duplex
 descrição 335
 usando um modem 337
Protocolo DF1 half-duplex
 descrição 336
protocolos de comunicação 59
 DF1 full-duplex 335
 DF1 half-duplex 336
PTO 143
 entradas/saídas configuráveis 146
 sinais de entrada/saída fixos 146
Pulso PTO 146, 147

R

receita 243
 erros do bloco de funções 245
 especificações 243
 estrutura do diretório 244
 parâmetros do bloco de funções 244
 status do bloco de funções 245
 tipos de dados 241
recomendações para fiação 47
Recursos adicionais 13
Recursos do hardware 16
registro de dados 237, 238
 diagrama de temporização 240
 especificações 239
 estrutura do diretório 238
 regras de execução 241
 tipos de dados 241
Regras para execução 138
regras para execução 238
relé de controle mestre 35
 chaves de parada de emergência 36
 usando esquema de símbolos ANSI/CSA 38
 usando esquema de símbolos IEC 37
retenção da variável 141
Retries 342, 347, 349
rotina de falha
 descrição da operação 300
 operação em relação ao programa principal de controle 297
 prioridade das interrupções 299
rotina de falha do usuário
 criando uma rotina de falha do usuário 300
 falhas recuperáveis e não recuperáveis 300

S

saída ativa
 regras gerais 153
saída do eixo
 regras gerais 152
segurança do controlador 221
senha do controlador 221
 recuperar 226
servidor Modbus/TCP 61
servo/inversor energizado 146, 147
servo/inversor pronto 146, 147
Servo-Drive 143
sinal de "em posição" 147
sincronização de tempo 126

- Sobre o controlador** 23
- soquetes cliente/servidor** 60, 64
- Status da rede** 315
- Status de alimentação** 314
- Status de comunicação serial** 314
- Status de force** 314
- Status de saída** 314
- status enable e valid**
 - regras gerais 153
- Supressão EOT** 342, 347, 349
- supressores de transiente**
 - para partidas de motor 49
 - uso 48
- supressores de transientes**
 - recomendados 49

T

- tabela de implementação** 131
- Taxa de comunicação** 341, 346, 348, 351
- Tempo limite de coleta** 342, 347
- transformadores de isolamento**
 - considerações de alimentação 34
- transmissão de comunicação CIP** 64

U

- Usando a função de interrupção selecionada em função do tempo (STI)** 306
- usando interrupções** 297
- utilizando chaves de parada de emergência** 36
- utilizando o contador de alta velocidade e chave de fim de curso programável**
 - 191

V

- validar endereço IP** 74
- varredura do programa** 238
- Verificar se as forças (travas) estão habilitadas** 289
- Visão geral da execução de programas** 137

Observações:

Suporte Rockwell Automation

Use estes recursos para acessar as informações de suporte.

Centro de suporte técnico	Encontre ajuda em vídeos tutoriais, perguntas frequentes, chats, fóruns de usuários e notificações de atualização de produtos.	rok.auto/support
Knowledgebase	Acesse artigos da Knowledgebase.	rok.auto/knowledgebase
Números de telefone do suporte técnico local	Localize o número de telefone do seu país.	rok.auto/phonesupport
Literature Library	Encontre instruções de instalação, manuais, folhetos e publicações de dados técnicos.	rok.auto/literature
Product Compatibility and Download Center (Centro de download e compatibilidade de produtos - PCDC)	Faça o download do firmware e os arquivos associados (como Add-on Profile, EDS e DTM) e acesse as notas da versão do produto.	rok.auto/pcdc

Comentários sobre a documentação

Seus comentários nos ajudarão a melhorar a documentação. Se você tiver alguma sugestão sobre como melhorar nosso conteúdo, preencha o formulário em rok.auto/docfeedback.

Resíduos dos equipamentos elétricos e eletrônicos (WEEE)



No fim do ciclo de vida, esses equipamentos devem ser coletados separadamente de qualquer lixo municipal não selecionado.

A Rockwell Automation mantém informações atuais de conformidade ambiental de produto em seu site em rok.auto/pec.

Allen-Bradley, CompactBlock LDX I/O, CompactLogix, Connected Components Workbench, ControlFLASH, ControlLogix, Data-Set, DH+, expanding human possibility, FactoryTalk, FactoryTalk Linx, FactoryTalk Linx Gateway, Kinetix, Micro800, Micro810, Micro820, Micro830, Micro850, Micro870, PanelView Component, PanelView Plus, PartnerNetwork, PowerFlex, Rockwell Automation, RSLinx, RSLinx Classic, RSLogix 500 e TechConnect são marcas comerciais da Rockwell Automation, Inc.

CIP, DeviceNet e EtherNet/IP são marcas comerciais de ODVA, Inc.

Microsoft, Excel, Visual Studio e Windows são marcas comerciais da Microsoft Corporation.

microSD é uma marca comercial da SD-3C.

As marcas comerciais que não pertencem à Rockwell Automation são propriedade de suas respectivas empresas.

Conecte-se conosco

rockwellautomation.com — expanding human possibility®

AMÉRICAS: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 EUA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444

EUROPA/ORIENTE MÉDIO/ÁFRICA: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Bélgica, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640

ÁSIA-PACÍFICO: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

BRASIL: Rockwell Automation do Brasil Ltda., Rua Verbo Divino, 1488 – 1º andar, Chac. Sto Antonio, 04719-904, São Paulo, SP, Tel: (55 11) 5189-9500,

www.rockwellautomation.com.br

PORTUGAL: Rockwell Automação, Lda., Av. Prof. Dr. Cavaco Silva, Edifício Ciência II, n.º 11 – 2ºC, Taguspark, Porto Salvo 2740-120, Tel.: (351) 214 225 500,

www.rockwellautomation.com.pt

Publicação 2080-UM002M-PT-E – Abril 2022

Copyright © 2022 Rockwell Automation, Inc. Todos os direitos reservados.